

EXHIBIT C

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Inventors: Mendez et al.

Patent No.: 7,225,231

Filed: Aug 1, 2001

For: System and Method for Transmitting
Workspace Elements Across a Network

REQUEST FOR REEXAMINATION UNDER
35 U.S.C. §§ 302-307 AND
37 C.F.R. § 1.510

Mail Stop Ex Parte Reexamination
ATTN: Central Reexamination Unit
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REQUEST FOR *EX PARTE* REEXAMINATION OF U.S. PATENT 7,225,231

TABLE OF CONTENTS

LIST OF EXHIBITS	7
A. PRIOR ART (PA).....	7
B. RELEVANT PATENT MATERIALS (PAT)	8
C. CLAIM CHARTS (CC)	8
D. OTHER DOCUMENTS (OTH)	9
I. OVERVIEW OF THE ‘231 PATENT AND SUBSTANTIAL NEW	
QUESTIONS OF PATENTABILITY.....	13
A. SUMMARY OF THE DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT	
OF THE ‘231 PATENT	13
B. ‘231 PATENT APPLICATION PROSECUTION HISTORY	15
C. OVERVIEW OF SUBSTANTIAL NEW QUESTIONS OF PATENTABILITY	18
D. SUMMARY OF THE PRIOR ART	19
1. BROWN.....	19
2. LAMB	20
3. LOTUS DEPLOYMENT GUIDE	21
4. LOTUS ADMIN GUIDE	23
5. LOTUS NOTES SYSTEM.....	25
6. BECKHARDT	25
7. SHAHEEN	26
8. WRIGHT	28
9. GLOMOP.....	30
10. LAMBERT	31
11. SATYANARAYANAN	31
12. KUMAR	33
13. KISTLER	34
14. CODA SYSTEM.....	36
E. RELATED REEXAMINATION PROCEEDINGS.....	36
F. RELATED CO-PENDING LITIGATION REQUIRES TREATMENT WITH SPECIAL	
DISPATCH AND PRIORITY OVER ALL OTHER CASES.....	37
G. CLAIM CONSTRUCTION.....	37

II.	STATEMENT UNDER 37 C.F.R. § 1.510(B) OF EACH SUBSTANTIAL NEW QUESTION OF PATENTABILITY	41
A.	ANTICIPATION BY BROWN	41
B.	ANTICIPATION BY LAMB	42
C.	ANTICIPATION BY LOTUS DEPLOYMENT GUIDE	42
D.	ANTICIPATION BY LOTUS ADMIN GUIDE.....	42
E.	OBVIOUSNESS OVER LOTUS NOTES	43
F.	ANTICIPATION BY BECKHARDT	43
G.	ANTICIPATION BY SHAHEEN	44
H.	ANTICIPATION BY WRIGHT.....	44
I.	ANTICIPATION BY GLOMOP	44
J.	ANTICIPATION BY LAMBERT.....	44
K.	ANTICIPATION BY SATYANARAYANAN	45
L.	ANTICIPATION BY KUMAR	45
M.	ANTICIPATION BY KISTLER	45
N.	OBVIOUSNESS OVER CODA	46
III.	CONCISE STATEMENT OF SUBSTANTIAL NEW QUESTION OF PATENTABILITY	46
A.	CLAIMS 1-6 ARE ANTICIPATED BY BROWN UNDER 35 U.S.C. § 102	47
1.	CLAIM 1	47
2.	CLAIM 2	51
3.	CLAIM 3	54
4.	CLAIM 4	58
5.	CLAIM 5	62
6.	CLAIM 6	66
B.	CLAIMS 1-6 ARE ANTICIPATED BY LAMB UNDER 35 U.S.C. § 102	70
1.	CLAIM 1	70
2.	CLAIM 2	75
3.	CLAIM 3	79
4.	CLAIM 4	84
5.	CLAIM 5	88
6.	CLAIM 6	93

C.	CLAIMS 1-6 ARE ANTICIPATED BY LOTUS DEPLOYMENT GUIDE UNDER 35 U.S.C. § 102.....	98
1.	CLAIM 1	98
2.	CLAIM 2	101
3.	CLAIM 3	103
4.	CLAIM 4	106
5.	CLAIM 5	109
6.	CLAIM 6	112
D.	CLAIMS 1-6 ARE ANTICIPATED BY LOTUS ADMIN GUIDE UNDER 35 U.S.C. § 102	115
1.	CLAIM 1	115
2.	CLAIM 2	118
3.	CLAIM 3	120
4.	CLAIM 4	123
5.	CLAIM 5	125
6.	CLAIM 6	128
E.	CLAIMS 1-6 ARE RENDERED OBVIOUS BY THE LOTUS NOTES SYSTEM UNDER 35 U.S.C. § 103.....	130
F.	CLAIMS 1-6 ARE ANTICIPATED BY BECKHARDT UNDER 35 U.S.C. § 102	131
1.	CLAIM 1	131
2.	CLAIM 2	135
3.	CLAIM 3	139
4.	CLAIM 4	144
5.	CLAIM 5	148
6.	CLAIM 6	152
G.	CLAIMS 1-6 ARE ANTICIPATED BY SHAHEEN UNDER 35 U.S.C. § 102	156
1.	CLAIM 1	156
2.	CLAIM 2	162
3.	CLAIM 3	168
4.	CLAIM 4	175
5.	CLAIM 5	180
6.	CLAIM 6	186
H.	CLAIMS 1-6 ARE ANTICIPATED BY WRIGHT UNDER 35 U.S.C. § 102	192

1.	CLAIM 1	192
2.	CLAIM 2	197
3.	CLAIM 3	202
4.	CLAIM 4	206
5.	CLAIM 5	211
6.	CLAIM 6	216
I.	CLAIMS 1-6 ARE ANTICIPATED BY GLOMOP UNDER 35 U.S.C. § 102	221
1.	CLAIM 1	221
2.	CLAIM 2	224
3.	CLAIM 3	225
4.	CLAIM 4	227
5.	CLAIM 5	229
6.	CLAIM 6	231
J.	CLAIMS 1-6 ARE ANTICIPATED BY LAMBERT UNDER 35 U.S.C. § 102	234
1.	CLAIM 1	234
2.	CLAIM 2	236
3.	CLAIM 3	237
4.	CLAIM 4	239
5.	CLAIM 5	241
6.	CLAIM 6	243
K.	CLAIMS 1-6 ARE ANTICIPATED BY SATYANARAYANAN UNDER 35 U.S.C. § 102	245
1.	CLAIM 1	245
2.	CLAIM 2	247
3.	CLAIM 3	249
4.	CLAIM 4	251
5.	CLAIM 5	253
6.	CLAIM 6	255
L.	CLAIMS 1-6 ARE ANTICIPATED BY KUMAR UNDER 35 U.S.C. § 102	257
1.	CLAIM 1	257
2.	CLAIM 2	260
3.	CLAIM 3	262
4.	CLAIM 4	264

5.	CLAIM 5	266
6.	CLAIM 6	268
M.	CLAIMS 1-6 ARE ANTICIPATED BY KISTLER UNDER 35 U.S.C. § 102	271
1.	CLAIM 1	271
2.	CLAIM 2	274
3.	CLAIM 3	276
4.	CLAIM 4	279
5.	CLAIM 5	281
6.	CLAIM 6	284
N.	CLAIMS 1-6 ARE RENDERED OBVIOUS BY THE CODA SYSTEM UNDER 35 U.S.C. § 103.....	286
IV.	EXPECTED ARGUMENTS AND AMENDMENTS.....	287
A.	LOTUS NOTES RUNNING ON A <i>SMART PHONE</i>	288
B.	PRIOR ART DISCLOSURE OF <i>SMART PHONE</i>	289
C.	LOTUS NOTES <i>TRANSLATION</i>	289
D.	LOTUS NOTES <i>SYNCHRONIZATION</i>	291
E.	<i>SYNCHRONIZING ON A DETECTED CHANGE OR PREDETERMINED NUMBER OF CHANGES</i>	292
F.	<i>FIREWALL</i>	294
G.	<i>TRANSLATION</i>	295
V.	CONCLUSION	296
	CERTIFICATE OF SERVICE	298

TABLE OF EXHIBITS

LIST OF EXHIBITS

The exhibits to the present Request are arranged in four groups: prior art (“PA”), relevant patent prosecution file history, patents, and claim dependency relationships (“PAT”), claim charts (“CC”), and other (“OTH”).

A. PRIOR ART (PA)

PA-SB08A	USPTO Form SB/08A
PA-A	Brown, Kevin et. al., <u>Mastering Lotus Notes</u> , 1995 Sybex Incorporated. (“Brown”)
PA-B	Lamb, John P. and Lew, Peter W., <u>Lotus Notes Network Design</u> , 1996 McGraw-Hill. (“Lamb”)
PA-C	<u>Lotus Notes Release 4 Deployment Guide</u> , 1995 Lotus Development Corporation, Cambridge, MA. (“Lotus Deployment Guide”)
PA-D	<u>Lotus Notes Release 4 Administrator’s Guide</u> , 1995 Lotus Development Corporation, Cambridge, MA. (“Lotus Admin Guide”)
PA-E	U.S. Patent No. 5,787,441 to Steven R. Beckhardt, Method Of Replicating Data At A Field Level, July 28, 1998 (“Beckhardt”).
PA-F	European Patent No. 0 684 558 to Amal Ahmed Shaheen, “Distributed Data Processing System,” filed March 31, 1995. (“Shaheen”)
PA-G	U.S. Patent Number 5,857,201 to Gerald V. Wright, Jr et al. (“Wright”)
PA-H	GloMop Group, <u>GloMop: Global Mobile Computing By Proxy</u> , Univ. of Ca. at Berkley (Sept.1995). (“GloMop”)
PA-I	Lambert M., <u>Pcmail: A Distributed Mail System for Personal Computers</u> , Network Working Group, June 1988. (“Lambert”)
PA-J	Mahadev Satyanarayanan, <u>Mobile Information Access</u> , IEEE Personal Communications (Feb. 1996). (“Satyanarayanan”)
PA-K	Kumar Puneet, <u>Mitigating the Effects of Optimistic Replication in a Distributed File System</u> , Ph.D. Thesis, School of Comp. Sci., Carnegie Mellon Univ. (Dec. 1994). (“Kumar”)

PA-L	Kistler, James, <u>Disconnected Operation in a Distributed File System</u> , Ph.D. thesis, Dept. of Comp. Sci., Carnegie Mellon Univ. (May 1993). (“Kistler”)
------	---

B. RELEVANT PATENT MATERIALS (PAT)

PAT-A	U.S. Patent No. 7,225,231 (the ‘231 patent)
PAT-B	File Wrapper for the ‘231 Patent

C. CLAIM CHARTS (CC)

CC-A	Claim Chart for Brown
CC-B	Claim Chart for Lamb
CC-C	Claim Chart for Lotus Deployment Guide
CC-D	Claim Chart for Lotus Admin Guide
CC-E	Claim Chart for Beckhardt
CC-F	Claim Chart for Shaheen
CC-G	Claim Chart for Wright
CC-H	Claim Chart for GloMop
CC-I	Claim Chart for Lambert
CC-J	Claim Chart for Satyanarayanan
CC-K	Claim Chart for Kumar
CC-L	Claim Chart for Kistler
CC-M	Claim Chart for Lotus Notes System
CC-N	Claim Chart for Coda System

D. OTHER DOCUMENTS (OTH)

- OTH-A Copy of the Amended Answer and Counter Claim filed by Visto Corp. in *Research In Motion, Ltd. v. Visto Corp.*, Civil Action No. 3:07-CV-03177-MMC (N.D. Ca.), filed Sept. 18, 2007.
- OTH-B Visto's Disclosure of Asserted Claims and Infringement Contentions under Patent Local Rule 3-1 in *Research In Motion Limited vs. Visto Corporation.*, 3:07-CV-03177-MMC (N.D. Ca.), filed Oct. 19, 2007.
- OTH-C Copy of Patent Rule 4-3 Joint Claim Construction and Pre-Hearing Statement in *Visto Corp. v. Research in Motion, Ltd.*, Civil Action No. 2:06-CV-181-TJW (E.D. Tex. filed September 4, 2007).
- OTH-D Copy of Visto's opening claim construction brief in *Visto Corp. v. Seven Networks, Inc.*, Civil Action No. 2:03-CV-333-TJW (E.D. Tex.), filed Apr. 20, 2005.
- OTH-E Copy of Visto's opening claim construction brief in *Visto Corp. v. Sproqit Technologies, Inc.*, Civil Action No. 3:04-CV-00651-EMC (N.D. Cal.), filed May 30, 2006.
- OTH-F Copy of Joint Claim Construction and Prehearing Statement of Plaintiff Visto Corp. and Defendant Smartner Information Systems, *Visto Corp. v. Smartner Information Systems, Ltd.*, Civil Action No. 2:05-CV-91-TJW (E.D. Tex.), filed Jul. 20, 2006.
- OTH-G Copy of Reexamination Office Action rejecting all claims of US. Patent No. 7,039,221 (Control No. 90/008,162).
- OTH-H Copy of Patent Owner's Response to the Reexamination Office Action rejecting all claims of US. Patent No. 7,039,221 (Control No. 90/008,162).
- OTH-I Copy of Court's Claim Construction in *Visto Corp. v. Seven Networks, Inc.*, Civil Action No. 2:03-CV-333-TJW (E.D. Tex. filed September 23, 2003).
- OTH-J Stephen Beckhardt's Testimony in *Visto Corp. v. Seven Networks, Inc.*, Civil Action No. 2:03-CV-333-TJW (E.D. Tex.), taken April 27, 2006.

OTH-K Visto's Reply Brief in Support of its Claim Construction, pp. 10, *Visto v. Sproquit Technologies*, Civil Action No. 3:04-cv-00651-EMC (N.D. Cal. June 16, 2006).

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Inventors: Mendez et al.

Patent No.: 7,225,231

Filed: Aug 1, 2001

For: System and Method for Transmitting
Workspace Elements Across a Network

REQUEST FOR REEXAMINATION UNDER
35 U.S.C. §§ 302-307 AND
37 C.F.R. § 1.510

Mail Stop Ex Parte Reexamination
ATTN: Central Reexamination Unit
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REQUEST FOR *EX PARTE* REEXAMINATION OF U.S. PATENT 7,225,231

Dear Sir:

The undersigned hereby respectfully requests reexamination pursuant to 35 U.S.C. §§ 302-307, 37 C.F.R. § 1.510, and MPEP §§ 2240(II) and 2295, of claims 1-6 of U.S. Patent No. 7,225,231 (“the ‘231 patent”) filed Aug 1, 2001 and issued May 29, 2007 to Daniel Mendez et al. (Exhibit PAT-A). Reexamination is requested in view of the substantial new questions of patentability (“SNQ”) presented below. Requester reserves all rights and defenses available including, without limitation, defenses as to invalidity and unenforceability. By simply filing this Request in compliance with the Patent Rules, Requester does not represent, agree or concur that the ‘231 patent is enforceable¹ and by asserting the SNQ herein, Requester specifically asserts that original claims 1-6 of the ‘231 patent are in fact not patentable and as such the U.S. Patent and Trademark Office (the

¹ As alleged by Patent Owner in the below defined Underlying Litigation, and as required by 37 C.F.R. § 1.510(a), the ‘231 patent is still within its period of enforceability for reexamination purposes, to the extent that the ‘231 patent has not lapsed for failure to pay maintenance fees, has not been the subject of any Terminal Disclaimer, and has not yet been held unenforceable in a court of competent jurisdiction.

“Office”) should reexamine and find all claims unpatentable and cancel all claims of the ‘231 patent, rendering the ‘231 patent null, void and otherwise unenforceable.

Requester notes that the ‘231 patent, for which reexamination is requested, was asserted by Visto Corp, (hereinafter “the Patent Owner”), in the litigation *Research In Motion Limited vs. Visto Corporation.*, Case No. C-07-3177 (N.D. Cal. June 15, 2007) (“the Pending Litigation”). See Patent Owner's First Amended Answer to Complaint and Counterclaims filed Sept. 18, 2007 at Exhibit OTH-A. For purposes of this Request, the Requester has and will construe all claim language from the claims asserted by the patentee in the litigation in the manner proffered by Visto Corp. Such statements by the patentee may be used by the Office to interpret claim language at issue.² When the claims are interpreted in the manner proffered by Visto Corp., or even under a narrower, more reasonable interpretation of the claims, the claims are unpatentable in view of the new prior art references presented herein.

By construing the claim language in the manner proffered by Visto Corp., and/or as otherwise set forth explicitly or implicitly herein, the Requester is not admitting and/or acquiescing as to the correctness and/or reasonableness of Visto Corp.'s proffered claim construction in the litigation and/or as otherwise set forth herein. In fact, the Requester expressly challenges, and will continue to vigorously challenge, Visto Corp.'s proffered claim construction, in whole or part, as the litigation continues.

² See 37 C.F.R. § 1.104(c)(3): “In rejecting claims the examiner may rely upon admissions by the applicant, or the patent owner in a reexamination proceeding, as to any matter affecting patentability[.]”

I. OVERVIEW OF THE '231 PATENT AND SUBSTANTIAL NEW QUESTIONS OF PATENTABILITY

A. SUMMARY OF THE DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT OF THE '231 PATENT

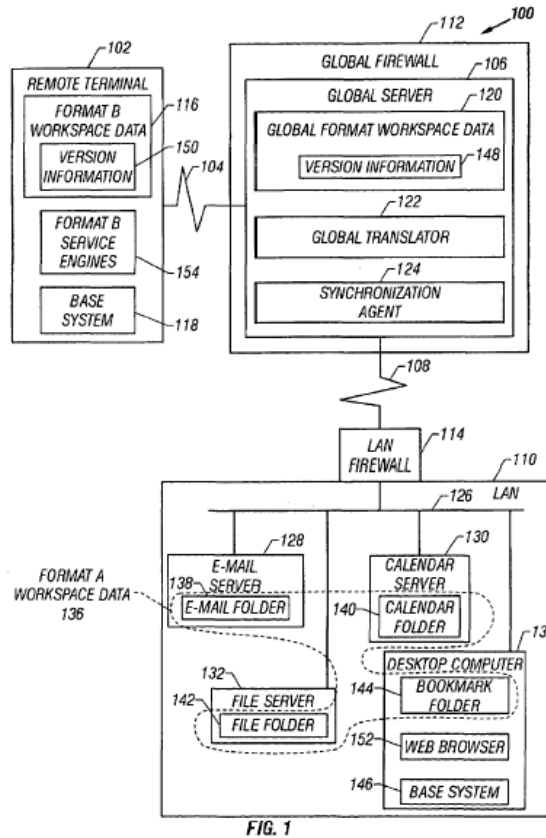


FIG. 1

Figure 1 of the '231 Patent

The '231 patent discloses in Figure 1 a global server that is connected to a LAN as well as to a remote terminal. '231 patent at Col. 2, line 65-Col. 3, line 6. The system stores data such as e-mail, files, calendar, etc., which are collectively referred to as workspace data. '231 patent at Col. 3, lines 13-20. The format of such workspace data may be the same among clients or may be different. '231 patent at Col. 3, lines 34-60. For example, workspace data stored on client may be in format "A," while workspace data stored on client may be in format "B." '231 patent at Col. 3, lines 34-60. The global server includes a "global translator" that translates workspace data between a "global" format as maintained for workspace data and the various other formats in which workspace data may be maintained on other devices. '231 patent at Col. 4, lines 4-27.

Workspace data is also stored on the global server, which includes independently modifiable copies of workspace data stored on the other devices. '231 patent at Col. 4, lines 4-27. The global server may be a single server or a network of multiple servers. Col. 11, lines 17-20.

According to the '231 patent, the workspace data 116 and 136 are independently synchronized with the workspace data on the global server, workspace data 120. '231 patent at Col. 5, lines 5-9. Therefore workspace data 116 and 136 are indirectly synchronized with each other. '231 patent at Col. 5, lines 5-9. The synchronization-start module 420 of the base system determines when to initiate synchronization of workspace data 116 or 136 with workspace data 120. '231 patent at Col. 7, lines 7-9.

The synchronization process of the preferred embodiment is charted in Fig. 7.

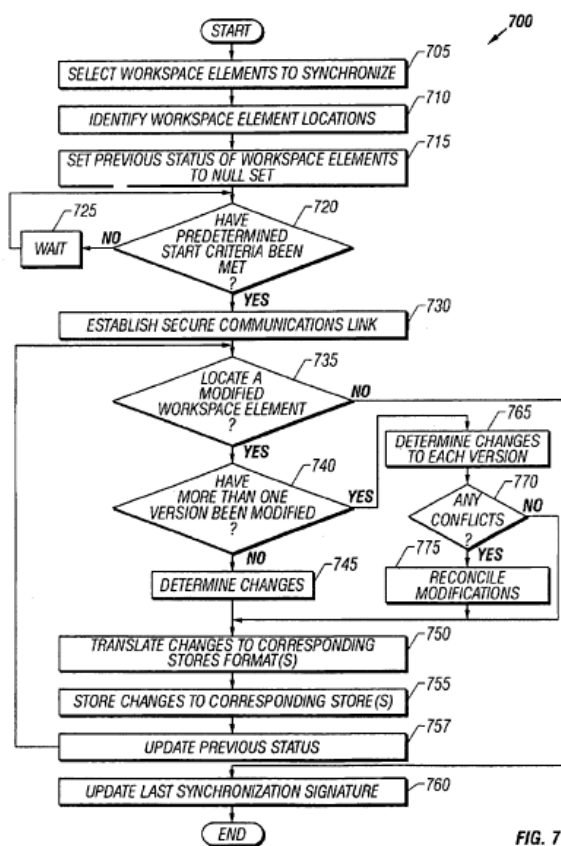


FIG. 7

Figure 7 of the '231 patent

The general synchronization modules 425 and 515 in step 735 determine whether any workspace data elements of workspace data have been modified in step 740 by comparing version date information of each selected workspace data element with the date of a last synchronization signature. ‘231 patent at Col. 9, lines 25-40. If only one workspace data has been modified, the changes are determined in step 745, translated in step 750, and sent to the other stores in step 755. ‘231 patent at Col. 9, line 48-Col. 10, line 3.

If more than one workspace data is determined to have changed in step 740, the content-based synchronization module 430 determines if there are any conflicts in the changes in step 770. ‘231 patent at Col. 10, lines 4-9. If there are no conflicts detected in step 770 then the changes in each version will be translated and forward to other stores. ‘231 patent at Col. 10, lines 13-17. Otherwise, the conflicting changes detected in step 770 will have to be reconciled in step 775. ‘231 patent at Col. 10, lines 17-23. Reconciliation includes requesting instructions from the user or reconciling based on preselected preferences. ‘231 patent at Col. 10, lines 19-23.

B. ‘231 PATENT APPLICATION PROSECUTION HISTORY

The ‘231 patent was filed as a continuation of the US Patent No. 6,708,221 (“the ‘221 patent”). See Exhibit PAT-B at Preliminary Amendment dated 07-01-01, pg.4. Patent Owner filed a preliminary amendment and canceled all existing claims and added six new claims, all independent claims, that after substantial amendments would results in the six claims issued as in the ‘231 patents. See Exhibit PAT-B at Preliminary Amendment dated 07-01-01, pg.1-4.

In the first office action, the examiner rejected all six claims in view of US Patent No. 5,974,238 to Chase and 5,333,266 to Boraz. See Exhibit PAT-B at Non-Final Office Action dated 05-17-04, pg.1-4. In response, the Patent Owner amended the claims to add limitations for transmitting information from a first store to a global server and from the global server to a second store. See Exhibit PAT-B at Patent Owner Amendment and Remarks dated 11-17-04. The first amendments made to claim 35 [issued as claim 1], reproduced below, are exemplary.

35. A method, comprising:

providing a first memory store for storing workspace elements;

providing a second memory store coupled via an electronic network to the first memory store for storing workspace elements copies;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new workspace elements at the first memory store; ~~and~~

electronically transmitting copies of the new workspace elements via the electronic network from the first memory store to a global server;
and

electronically transmitting copies of the new workspace element elements via the electronic network from the global server to the second memory store.

See Exhibit PAT-B at Patent Owner Amendment and Remarks dated 11-17-04.

Next the Office rejected all the claims as obvious over the '221 patent. See Exhibit PAT-B at Non-Final Office Action dated 03-29-05. In response the Patent Owner filed a terminal disclaimer and the Examiner subsequently allowed the claims. See Exhibit PAT-B at Terminal Disclaimer dated 10-03-05 and Notice of Allowance dated 12-05-05. However, in response to a post allowance IDS filed by Patent Owner, the Examiner withdrew the allowance and rejected all the claims under 102(e) and 103(a) in view of US Patent No. 5,778,389 to Pruett. See Exhibit PAT-B at Non-Final Office Action dated 05-04-06.

In Response to the office action, Patent Owner substantially amended the claims to overcome Pruett. See Exhibit PAT-B at Patent Owner Amendment and Remarks dated 11-13-06. The final amendments made to claim 35 [issued as claim 1], reproduced below, are exemplary.

35. A method, comprising:

providing a first memory store including a first folder for storing first record entries in a file ~~workspace elements;~~

providing a second memory store coupled ~~via an electronic network~~ to the first memory store, the second memory store including a

second folder for storing second record entries in a file ~~workspace elements~~, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time ~~copies~~;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file ~~workspace elements~~ at the first memory store at a second moment in time later than the first moment in time;

electronically ~~synchronizing the first folder and the third folder, such that a new third record entry ~~workspace element~~ is created in the third folder based on transmitting copies of the new first record entries ~~workspace elements~~ via the electronic network from the first memory store to a global server;~~ and

electronically ~~transmitting copies~~ synchronizing the third folder and the second folder, such that a new ~~of the second record entry new workspace element~~ is created in the second folder based on the new third record entry ~~elements via the electronic network from the global server to the second memory store.~~³

See Exhibit PAT-B at Patent Owner Amendment and Remarks dated 11-13-06.

In addition to the amendments, Patent Owner also argued, “Moreover, Pruett et al. is directed to only copying entire files as contrasted with record entries in a file. [See, Pruett et al. at col. 2 ll. 18-19] Therefore, the reference does not anticipate [issued claim 1].”⁴ *Id.* Patent Owner states that the other claims are allowable for the same reasons. *Id.*

³ Requester notes that neither the terms ‘record’ nor ‘record entries’ are used in the specification of the ‘231 patent. Patent Owner alleges the support for these terms can be found at the ‘231 patent at col. 3, ll. 13-33, col. 4, ll. 4-39, and col. 5, ll. 5-9. Exhibit PAT-B at Patent Owner Amendment and Remarks dated 11-13-06. The smallest data structure disclosed is a “workspace element” and the smallest documents disclosed as workspace elements are entire files such as an e-mail, file, calendar, or bookmark. *Id.* There is no disclosure in the ‘231 patent of a data structure smaller than a “workspace element” which is comprised of an entire file at a minimum. Although Requester believes the terms are indefinite, Requester is unable to assert indefiniteness in a reexamination request and therefore will try to interpret these claim elements in accordance with claim construction standards set forth herein.

⁴ Although Patent Owner made clear during prosecution to overcome Pruett that “record entries in a file” is different than an “entire file”, in litigation the Patent Owner has asserted that this limitation is infringed by documents comprising an entire file. See Exhibit OTH-B, Visto’s Disclosure of Asserted Claims and Infringement Contentions under Patent Local Rule 3-1 at pg. 3. Specifically, Patent Owner claims the following types of documents infringe the “record entries in a file” limitation of the ‘231 patent: “emails,

The Office subsequently allowed the claims but did not state any express reasons for the allowance. See Exhibit PAT-B at Notice of Allowance dated 12-11-06.

Therefore, the Requester will look to the final amendments and arguments made by the Patent Owner to overcome Pruetz as the reasons the claims were allowed. As discussed below, all the prior art submitted in this request discloses all the limitations of the claims as issued, including the three folders for storing record entries of a file and synchronization of the record entries in the folders.

C. OVERVIEW OF SUBSTANTIAL NEW QUESTIONS OF PATENTABILITY

For the reasons set forth below, a SNQ as to all claims 1-6 is raised by the prior art references with respect to anticipation under 35 U.S.C. § 102. A complete listing of the prior art submitted with this Request is provided in the attached Form PTO-SB/08A, together with a full copy of each listed document.

The '231 patent contains six claims, all independent. The claims of the '231 patent are directed to a method and for *synchronizing*⁵ data relating to *record entries* in a file involving three *stores*. The *first store synchronizes* with the *third store* and then the *third store synchronizes* with the *second store* so that data contained in the *first store* and *second store* is consistent. Independent claim 1 of the '231 patent is exemplary and reads as follows.

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

contacts, calendar entries, notes, tasks, etc.” *Id.* Despite Patent Owner’s attempt to change the meaning of the term “record entries in a file” to improve their litigation position, the prior art presented in this reexamination request anticipates under either construction.

⁵Italicized terms are understood to incorporate by reference the patentee Visto Corporation’s own claim construction as discussed in further detail below and/or in a manner that are likely to be proffered by the patentee given its infringement contentions in the underlying litigation and its claim constructions in related litigation.

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

D. SUMMARY OF THE PRIOR ART

1. BROWN

The Brown reference was published in 1995 by Sybex Incorporated. Brown constitutes effective prior art under 35 U.S.C. §102.

Brown discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Brown patent discloses a *first memory store* (e.g. computer workstation) including a first folder for storing first *record entries* in a file (e.g. folders, documents, replicas, database). Brown at pp. 8, 40, 571. Brown also discloses a *second memory store* (e.g. another computer on the network) including a second folder for storing *record entries* in a file (e.g. folders, documents, replicas, database), and the second folder being *synchronized* (e.g. replication) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Brown at pp. 8, 40, 571, 569. Brown also discloses a *global server* (e.g. Lotus notes server) including a *third memory store*. Brown at p. 8.

Furthermore, Brown discloses that new first *record entries* in a file are received (e.g. changes to a document) at the *first memory store* at a moment in time later than the first moment in time. Brown at pp. 94, 582-83. Still further, Brown discloses electronically *synchronizing* the first folder and the third folder (e.g. replication, "update the shared database"), such that a new third *record entry* is created in the third folder

based on the new first *record entries*. Brown at pp. 571-72. Moreover, Brown additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. replication to User workstation computers), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Brown at pp. 571-72.

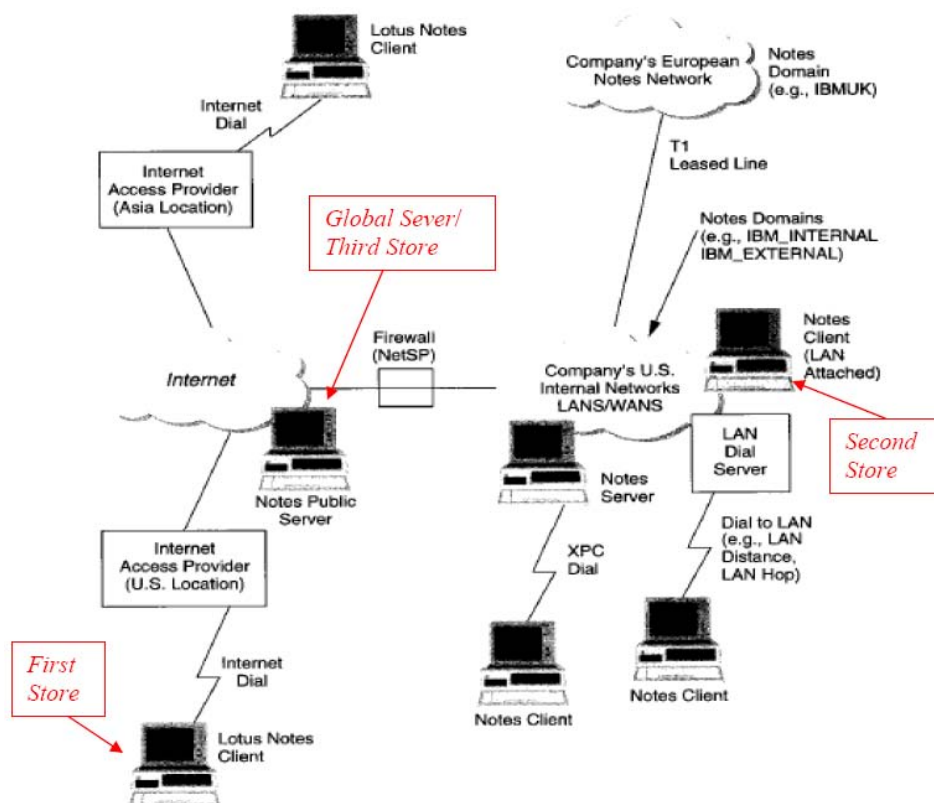
Brown was not before the Examiner during the prosecution of the '231 patent and has not been considered by the Office with regards to the '231 patent.

2. LAMB

The Lamb reference was published in 1996 by McGraw-Hill. Lamb constitutes effective prior art under 35 U.S.C. §102.

Lamb discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Lamb patent discloses a *first memory store* (e.g. Lotus notes client computer, laptops or notebooks, personal digital assistants (PDAs)) including a first folder for storing first *record entries* in a file (e.g. Lotus notes database, documents, updateable fields in documents). Lamb at pp. 7, 28, 178 at Fig. 2.1. Lamb also discloses a *second memory store* (e.g. another client computer on the network) including a second folder for storing *record entries* in a file (e.g. Lotus notes database, documents, updateable fields in documents), and the second folder being *synchronized* (e.g. replication) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Lamb at pp. 7, 28, 178 at Fig. 2.1. Lamb also discloses a *global server* (e.g. "hub server") including a *third memory store*. Lamb at p. 27, 128.

Furthermore, Lamb discloses that new first *record entries* in a file are received (e.g. changes to a document, fields, mail, databases) at the *first memory store* at a moment in time later than the first moment in time. Lamb at pp. 5-6, 28. Still further, Lamb discloses electronically *synchronizing* the first folder and the third folder (e.g. replication to hub server), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lamb at pp. 9, 28. Moreover, Lamb additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. replication to spoke), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Lamb at pp. 9, 28.



Lamb Fig. 2.1

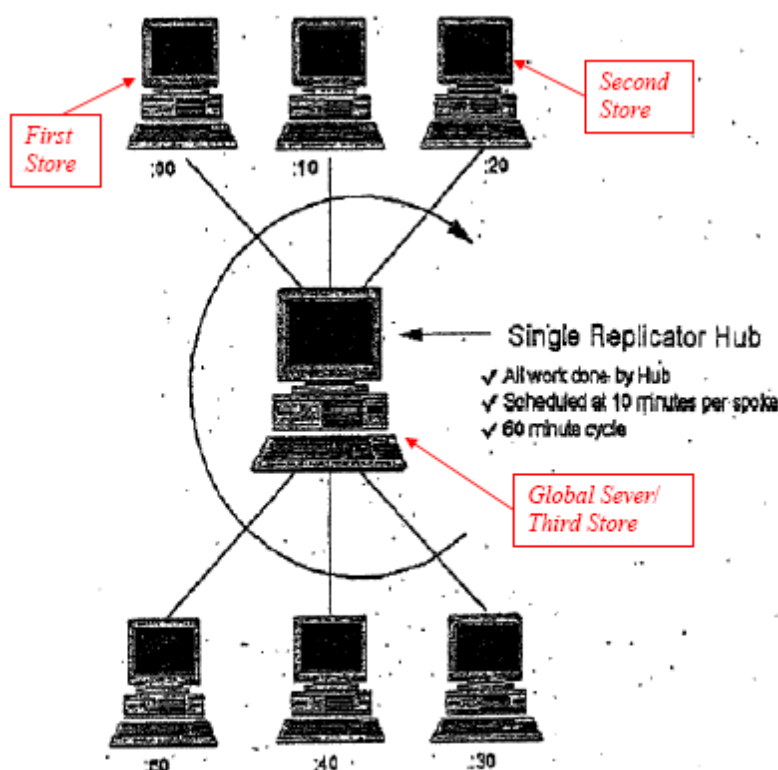
Lamb is cited on the face of the '231 patent. However, Lamb is being presented in a new light under MPEP 2242 (II) in view of the new arguments presented below. There was no statement by the Patent Owner to explain the pertinence of Lamb to the claims of the '231 patent to help the examiner sort through all of the submitted references. The Examiner did not base any rejection or make any comments regarding Lamb. The arguments below help explain how Lamb relates to the claims of the '231 patent in a light not previously considered by the examiner.

3. LOTUS DEPLOYMENT GUIDE

Lotus Deployment Guide was published in 1995 by Lotus Development Corporation. Lotus Deployment Guide constitutes effective prior art under 35 U.S.C. § 102.

Lotus Deployment Guide discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Lotus Deployment Guide patent discloses a *first memory store* (e.g. a server storing a

database) including a first folder for storing first *record entries* in a file (e.g. fields within documents of the database). Lotus Deployment Guide at p. 41. Lotus Deployment Guide also discloses a *second memory store* (e.g. second server storing a replica of the database) including a second folder for storing *record entries* in a file (e.g. fields within documents of the database), and the second folder being *synchronized* (e.g. replication) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Lotus Deployment Guide at p. 41, 51, figure on page 51. Lotus Deployment Guide also discloses a *global server* (e.g. hub server) including a *third memory store*. Lotus Deployment Guide at p 48.



Lotus Deployment Guide Figure on p. 51.

Furthermore, Lotus Deployment Guide discloses that new first *record entries* in a file are received (e.g. *record entries* on a server may be added to, updated, or deleted) at the *first memory store* at a moment in time later than the first moment in time. Lotus Deployment Guide at p. 41. Still further, Lotus Deployment Guide discloses electronically *synchronizing* the first folder and the third folder (e.g. Hub server replicates only the modified fields within a document with the other servers in sequential order,

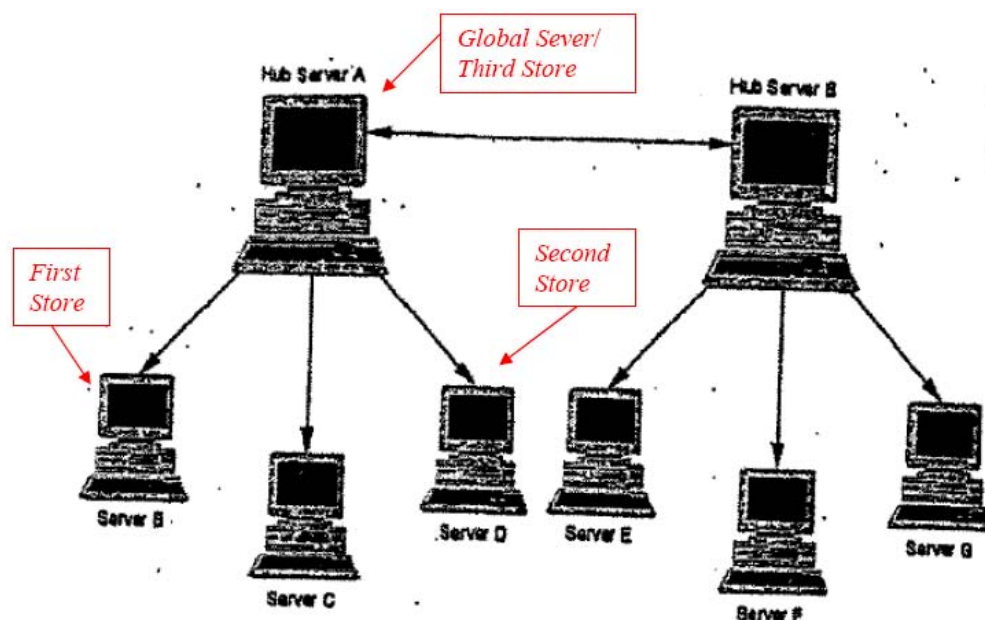
starting with the *first store*), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lotus Deployment Guide at pp. 41 and 51. Moreover, Lotus Deployment Guide additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. replication of the *second store* occurring after the hub server replicated with the *first store*), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Lotus Deployment Guide at pp. 41 and 51.

Lotus Deployment Guide is cited on the face of the '231 patent. However, Lotus Deployment Guide is being presented in a new light under MPEP 2242 (II) in view of the new arguments presented below. There was no statement by the Patent Owner to explain the pertinence of Lotus Deployment Guide to the claims of the '231 patent to help the examiner sort through all of the submitted references. The Examiner did not base any rejection or make any comments regarding Lotus Deployment Guide. The arguments below help explain how Lotus Deployment Guide relates to the claims of the '231 patent in a light not previously considered by the examiner.

4. LOTUS ADMIN GUIDE

Lotus Admin Guide was published in 1995 by Lotus Development Corporation. Lotus Deployment Guide constitutes effective prior art under 35 U.S.C. § 102.

Lotus Admin Guide discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Lotus Admin Guide patent discloses a *first memory store* (e.g. server) including a first folder for storing first *record entries* in a file (e.g. modifiable fields within a document). Lotus Admin Guide at 78. Lotus Admin Guide also discloses a *second memory store* (e.g. second server) including a second folder for storing *record entries* in a file (e.g. modifiable fields within a document), and the second folder being *synchronized* (e.g. replication) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Lotus Admin Guide at 71. Lotus Deployment Guide also discloses a *global server* (e.g. hub server) including a *third memory store*. Lotus Admin Guide at p. 77-78, see Figure on p. 78.



Lotus Admin Guide Figure on p. 78.

Furthermore, Lotus Admin Guide discloses that new first *record entries* in a file are received (e.g. users are able to later add, edit, or delete documents in a database) at the *first memory store* at a moment in time later than the first moment in time. Lotus Admin Guide at p. 72. Still further, Lotus Admin Guide discloses electronically *synchronizing* the first folder and the third folder (e.g. hub server replicates with spoke servers sequentially starting with the *first store*), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lotus Admin Guide at p. 77. Moreover, Lotus Admin Guide additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. hub server replicates with spoke servers sequentially), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Lotus Admin Guide at p. 77.

Lotus Admin Guide is cited on the face of the '231 patent. However, Lotus Admin Guide is being presented in a new light under MPEP 2242 (II) in view of the new arguments presented below. There was no statement by the Patent Owner to explain the pertinence of Lotus Admin Guide to the claims of the '231 patent to help the examiner sort through all of the submitted references. The Examiner did not base any rejection or make any comments regarding Lotus Admin Guide. The arguments below help explain

how Lotus Admin Guide relates to the claims of the '231 patent in a light not previously considered by the examiner.

5. LOTUS NOTES SYSTEM

Brown, Lamb, Lotus Deployment Guide and Lotus Admin Guide all discuss the Lotus Notes system. All four references describe with particularity the Lotus Notes system developed by IBM for synchronizing data among server and client computers. Since all of the above cited references pertain to Lotus Notes, then one of ordinary skill in the art would have looked through each prior art reference for necessary teachings for using at least Lotus Notes Revision 4. Each of the prior art references discloses either the same or additional features and configuration for use with Lotus Notes.

Brown was not before the Examiner during the prosecution of the '231 patent and has not been considered by the Office with regards to the '231 patent. Lamb, Lotus Deployment Guide and Lotus Admin Guide are cited on the face of the '231 patent but they were not commented on by the Examiner or used in any rejection during prosecution of the '231 patent. Therefore, this combination of art describing the Lotus Notes system has not been before and has not been considered by the Office with regards to the '231 patent.

6. BECKHARDT

Beckhardt issued July 28, 1998 from application No. 584,958, filed January 11, 1996. Beckhardt constitutes effective prior art under 35 U.S.C. § 102.

Beckhardt discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Beckhardt patent discloses a *first memory store* (e.g. at least one of the multiple computer systems on the network) including a first folder for storing first *record entries* in a file (e.g. document, record, file, text or other data). Beckhardt at Col. 1, lines 5-17, Col. 2, lines 3-19. Beckhardt also discloses a *second memory store* (e.g. another client computer on the network) including a second folder for storing *record entries* in a file (e.g. document, record, file, text or other data), and the second folder being *synchronized* (e.g. replication) with the first folder at a first moment in time, such that the content of the

second folder is consistent with the content of the first folder. Beckhardt at Col. 1, lines 5-17, Col. 2, lines 3-19. Beckhardt also discloses a *global server* (e.g. server holding central or master copy) including a *third memory store*. Beckhardt at Col. 1, lines 24-32.

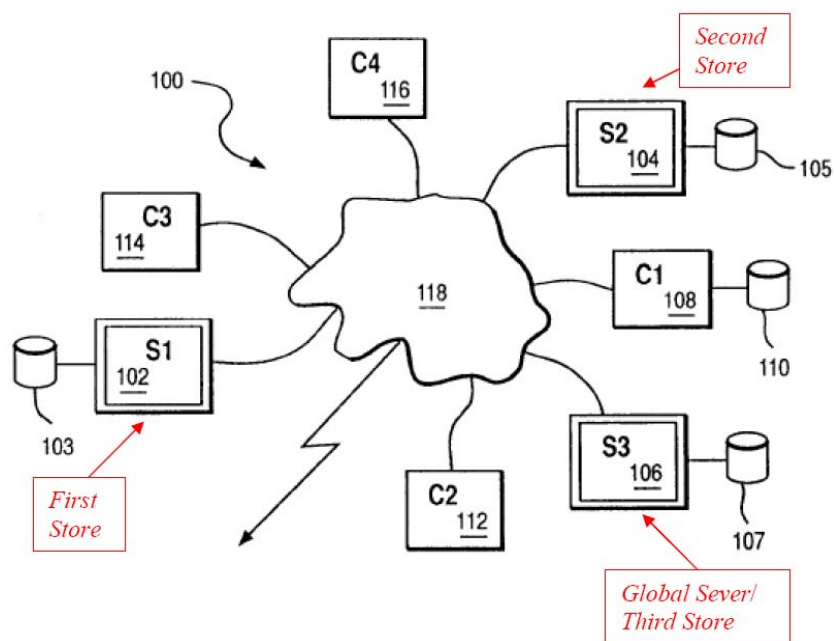
Furthermore, Beckhardt discloses that new first *record entries* in a file are received (e.g. changes made by User to document, record, file, text or other data) at the *first memory store* at a moment in time later than the first moment in time. Beckhardt at Col. 1, lines 24-32. Still further, Beckhardt discloses electronically *synchronizing* the first folder and the third folder (e.g. “Users then replicate changes they make in their document replicas to the master document”), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Beckhardt at Col. 1, lines 24-32. Moreover, Beckhardt additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. “the master is used to pass along these changes to the other working replicas of the document”), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Beckhardt at Col. 1, lines 24-32.

Beckhardt was not before the Examiner during the prosecution of the ‘231 patent and has not been considered by the Office with regards to the ‘231 patent.

7. SHAHEEN

Shaheen was filed on March 31, 1995 as EP 0 684 558 based on application 95302184.7. Shaheen constitutes effective prior art under 35 U.S.C. § 102.

Shaheen discloses all of the limitations in the claims of the ‘231 patent including the limitations the Patent Owner added for allowance. Specifically, the Shaheen patent discloses a *first memory store* (e.g. one of the servers in the fileset storage group (FSG)); also magnetic hard disks, optical disks or similar technology) including a first folder for storing first *record entries* in a file (e.g. fileset replica, modification log). Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1.



Shaheen Figure 1.

Shaheen also discloses a *second memory store* (e.g. another server in the FSG; also magnetic hard disks, optical disks or similar technology) including a second folder for storing *record entries* in a file (e.g. fileset replica, modification log), and the second folder being *synchronized* (e.g. replication; transmission of updates) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Shaheen also discloses a *global server* (e.g. “coordinator”) including a *third memory store*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58.

Furthermore, Shaheen discloses that new first *record entries* in a file are received (e.g. changes to a fileset or replica; changes to a modification log) at the *first memory store* at a moment in time later than the first moment in time. Shaheen at Col. 7, line 22-26. Still further, Shaheen discloses electronically *synchronizing* the first folder and the third folder (e.g. Coordinator requests, receives and merges logs from subordinates), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. Moreover, Shaheen additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. Coordinator sends the merged log to all the subordinates), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Shaheen at Col. 1, lines 24-32.

Shaheen was not before the Examiner during the prosecution of the '231 patent and has not been considered by the Office with regards to the '231 patent.

8. WRIGHT

Wright issued January 5, 1999 from application No. 665,422, filed June 8, 1996. Wright constitutes effective prior art under 35 U.S.C. § 102.

Wright discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Wright patent discloses a *first memory store* (e.g. FL client; handheld devices, personal digital assistants (PDA), as well as other portable devices such as laptops) including a first folder for storing first *record entries* in a file (e.g. database, email, a data, or work orders). Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Wright also discloses a *second memory store* (e.g. another client computer in the network; handheld devices, personal digital assistants (PDA), as well as other portable devices such as laptops) including a second folder for storing *record entries* in a file (e.g. database, email, a data, or work orders), and the second folder being *synchronized* (e.g. “synchronizing”) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Wright at Col. 5, lines 16-24, 46-52, Col. 6, lines 10-21; Fig. 2. Wright also discloses a *global server* (e.g. “FL server”, “gateway”) including a *third memory store*. Wright at Col. 5, lines 16-24, 46-52, Col. 6, lines 10-21; Fig. 2.

Furthermore, Wright discloses that new first *record entries* in a file are received (e.g. “full local database implementation that allows data to be manipulated and collected by the FL client”) at the *first memory store* at a moment in time later than the first moment in time. Wright at Col. 5, lines 46-52. Still further, Wright discloses electronically *synchronizing* the first folder and the third folder (e.g. “FL server 132 is responsible for manipulation of the FL client database 172, including retrieving data that has been collected by the client”), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Wright at Col. 5, lines 63-66. Moreover, Wright additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. synchronization of the two databases is performed), such that a

new second *record entry* is created in the second folder based on the new third *record entry*. Wright at Col. 5, lines 52-59.

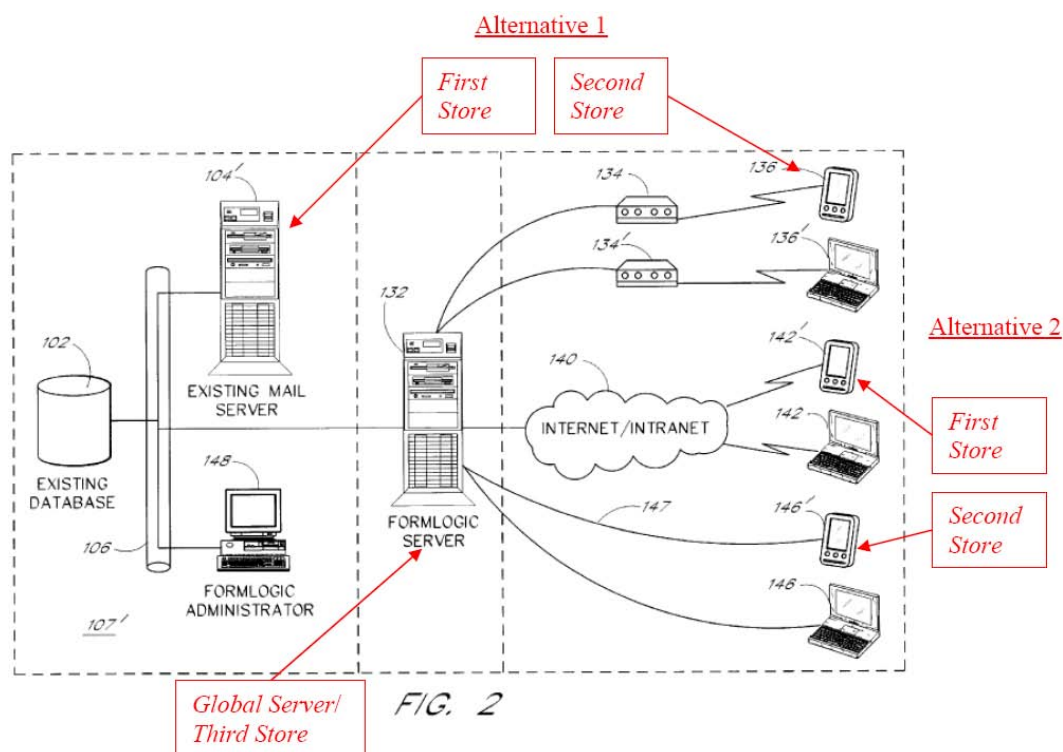


Figure 2 of the Wright Patent

Although the Wright Patent presented in this request is not cited on the face of the '231 patent, a continuation of the Wright patent is on the face of the patent. However, the Wright patent is being presented in a new light under MPEP 2242 (II) in view of the new arguments presented below. There was no statement by the Patent Owner to explain the pertinence of the continuation of the Wright patent to the claims of the '231 patent to help the examiner sort through all of the submitted references. The Examiner did not base any rejection or make any comments regarding the continuation of the Wright patent. The arguments below help explain how the Wright patent relates to the claims of the '231 patent in a light not previously considered by the examiner.

9. GLOMOP

Glomop was published on September 1995 by Univ. of Ca. at Berkley. Glomop constitutes effective prior art under 35 U.S.C. § 102.

Glomop discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Glomop patent discloses a *first memory store* (e.g. client computer, such as for example intel x86, MacIntosh PowerBook) including a first folder for storing first *record entries* in a file (e.g. folders, documents). GloMop at p. 5, § 1.6. GloMop also discloses a *second memory store* (e.g. another client computer, such as for example, intel x86 or MacIntosh PowerBook in the network) including a second folder for storing *record entries* in a file (e.g. fileset replica, modification log), and the second folder being *synchronized* (e.g. "The user synchronizes the file cache...") with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. GloMop at p. 7, § 2.1. GloMop also discloses a *global server* (e.g. proxy server) including a *third memory store*. GloMop at p. 2, § 1.2.

Furthermore, GloMop discloses that new first *record entries* in a file are received (e.g. new mail) at the *first memory store* at a moment in time later than the first moment in time. GloMop at p. 10, § 3.5. Still further, GloMop discloses electronically *synchronizing* the first folder and the third folder (e.g. sending a log file to resynchronize the user's inbox on the proxy with the one on the client), such that a new third *record entry* is created in the third folder based on the new first *record entries*. GloMop at p. 10, § 3.5. Moreover, GloMop additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. second client would download from the proxy server the new email uploaded by the first client to the proxy server), such that a new second *record entry* is created in the second folder based on the new third *record entry*. GloMop at p. 7, § 2.1.

GloMop was not before the Examiner during the prosecution of the '231 patent and has not been considered by the Office with regards to the '231 patent.

10. LAMBERT

Lambert was published in June 1988 by Network Working Group. Lambert constitutes effective prior art under 35 U.S.C. § 102.

Lambert discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Lambert patent discloses a *first memory store* (e.g. resource limited client) including a first folder for storing first *record entries* in a file (e.g. mail). Lambert at pp. 1-2. Lambert also discloses a *second memory store* (e.g. another resource limited client) including a second folder for storing *record entries* in a file (e.g. mail), and the second folder being *synchronized* (e.g. first and second clients are *synchronized*) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Lambert at p. 1-2, 18. Lambert also discloses a *global server* (e.g. resource rich *global server*) including a *third memory store*. Lambert at pp. 1-2.

Furthermore, Lambert discloses that new first *record entries* in a file are received (e.g. "update list" of messages that have been received at the first client since the last synchronization) at the *first memory store* at a moment in time later than the first moment in time. Lambert at p. 18. Still further, Lambert discloses electronically *synchronizing* the first folder and the third folder (e.g. first client will synchronize the new changes with the repository), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lambert at p. 18. Moreover, Lambert additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. repository synchronizes the changes with the second client), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Lambert at p. 18.

Lambert was not before the Examiner during the prosecution of the '231 patent and has not been considered by the Office with regards to the '231 patent.

11. SATYANARAYANAN

Satyanarayanan was published in June February 1996 by IEEE Personal Communications. Satyanarayanan constitutes effective prior art under 35 U.S.C. § 102.

Satyanarayanan discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Satyanarayanan patent discloses a *first memory store* (e.g. client, such as an IBM ThinkPad, contains a cache manager, Venus,) including a first folder for storing *first record entries* in a file (e.g. volumes are files and the volume objects). Satyanarayanan at 28, 30. Satyanarayanan also discloses a *second memory store* (e.g. second client with a cache manager, Venus, to store data) including a second folder for storing *record entries* in a file (e.g. volumes are files and the volume objects), and the second folder being *synchronized* (e.g. first and second clients synchronize with each other through the server) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Satyanarayanan at pp. 28 and 30. Satyanarayanan also discloses a *global server* (e.g. file servers mapped to a single namespace) including a *third memory store*. Satyanarayanan at pp. 27-28.

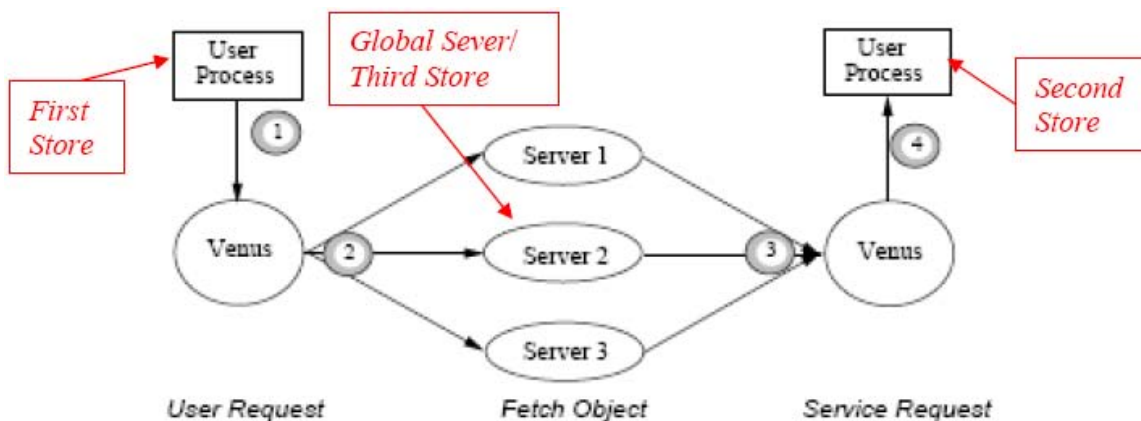
Furthermore, Satyanarayanan discloses that new *first record entries* in a file are received (e.g. first client logs changes to the data stored on it with volume stamps) at the *first memory store* at a moment in time later than the first moment in time. Satyanarayanan at p. 29. Still further, Satyanarayanan discloses electronically *synchronizing* the first folder and the third folder (e.g. first client reconnects to the Coda system, it resynchronizes), such that a new *third record entry* is created in the third folder based on the new *first record entries*. Satyanarayanan at p. 28. Moreover, Lambert additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. second client synchronizes with the file server and integrates any new changes on the file server), such that a new *second record entry* is created in the second folder based on the new *third record entry*. Satyanarayanan at p. 28.

Although Satyanarayanan is cited on the face of the '231 patent, Satyanarayanan is being presented in a new light under MPEP 2242 (II) in view of the new arguments presented below. There was no statement by the Patent Owner to explain the pertinence of Satyanarayanan to the claims of the '231 patent to help the examiner sort through all the submitted references. The Examiner did not base any rejection or make any comments regarding Satyanarayanan.

12. KUMAR

Kumar was published in December 1994 as a Thesis from Carnegie Mellon Univ. Kumar constitutes effective prior art under 35 U.S.C. § 102.

Kumar discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Kumar patent discloses a *first memory store* (e.g. client) including a first folder for storing first *record entries* in a file (e.g. directories and files). Kumar at pp. 7-8. Kumar also discloses a *second memory store* (e.g. second client with a cache manager, Venus, to store data) including a second folder for storing *record entries* in a file (e.g. directories and files), and the second folder being *synchronized* (e.g. first and second clients synchronize with each other through servers) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Kumar at 7, 8 and Fig. 2.2(a). Kumar also discloses a *global server* (e.g. one server of the set of servers that make up Vice) including a *third memory store*. Kumar at pp. 7-8 and Fig. 2.2(a).



Kumar Fig 2.2(a)

Furthermore, Kumar discloses that new first *record entries* in a file are received (e.g. file systems at the clients can be later updated) at the *first memory store* at a moment in time later than the first moment in time. Kumar at p. 8. Still further, Kumar discloses electronically *synchronizing* the first folder and the third folder (e.g. when data is updated

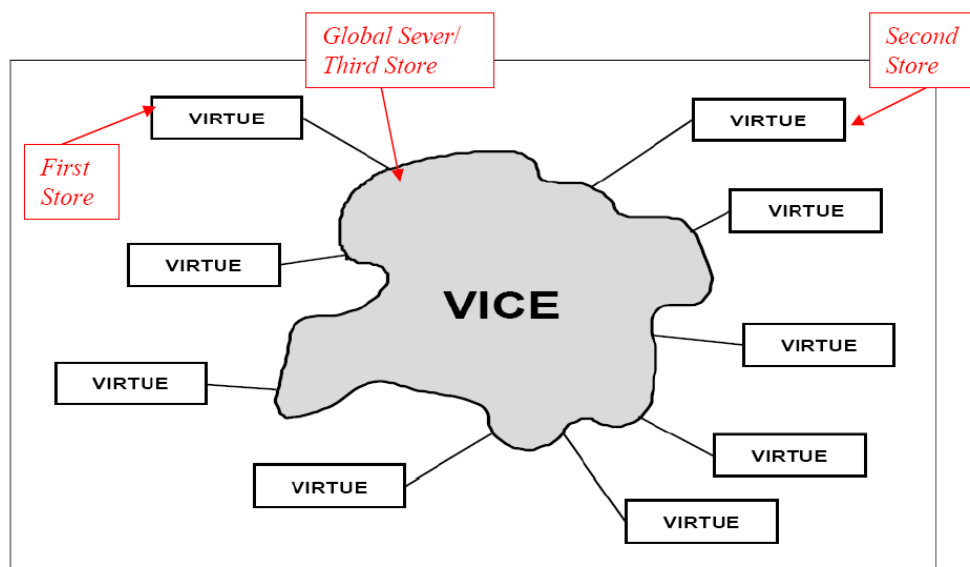
at the first client, it is sent to and *synchronized* with the servers), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Kumar at p. 8. Moreover, Kumar additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. clients with the old data synchronize with the servers to obtain the new updated data), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Kumar at 8.

Kumar was not before the Examiner during the prosecution of the '231 patent and has not been considered by the Office with regards to the '231 patent.

13. KISTLER

Kistler was published in May 1993 as a Thesis from Carnegie Mellon Univ. Kistler constitutes effective prior art under 35 U.S.C. § 102.

Kistler discloses all of the limitations in the claims of the '231 patent including the limitations the Patent Owner added for allowance. Specifically, the Kistler patent discloses a *first memory store* (e.g. caches on user workstations called Virtue that maintain folders in the file system) including a first folder for storing first *record entries* in a file (e.g. object descriptors). Kistler at pp. 14-18 and Fig. 2.1. Kistler also discloses a *second memory store* (e.g. second Virtue workstation) including a second folder for storing *record entries* in a file (e.g. object descriptors), and the second folder being *synchronized* (e.g. synchronize by updating caches) with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder. Kistler at 17-19 and Fig. 2.1. Kistler also discloses a *global server* (e.g. Vice) including a *third memory store*. Kistler at p. 14 and Fig 2.1.



The structure labeled "Vice" is a collection of trusted servers and untrusted networks. The nodes labeled "Virtue" are private or public workstations (or occasionally time-sharing systems). Vice is reputed to stand for "Vast Integrated Computing Environment," and Virtue for "Virtue is Realized through Unix and Emacs."

Kistler Fig. 2.1 at pg. 14.

Furthermore, Kistler discloses that new first *record entries* in a file are received (e.g. client may update cached objects) at the *first memory store* at a moment in time later than the first moment in time. Kistler at p. 19. Still further, Kistler discloses electronically *synchronizing* the first folder and the third folder (e.g. when an object is updated on a client, the updated object is *synchronized* the *global server*, Vice), such that a new third *record entry* is created in the third folder based on the new first *record entries*. Kistler at p. 18. Moreover, Kistler additionally discloses electronically *synchronizing* the third folder and the second folder (e.g. the *global server* and the outdated client synchronizes so the client gets the updated objects), such that a new second *record entry* is created in the second folder based on the new third *record entry*. Kistler at p. 18-19.

Kistler was not before the Examiner during the prosecution of the '231 patent and has not been considered by the Office with regards to the '231 patent.

14. CODA SYSTEM

Satyanarayanan, Kumar, and Kistler all discuss the Lotus Notes system. All three references describe with particularity the Coda system developed at Carnegie Mellon University for synchronizing data between server and client computers. Since all of the above cited references pertain to Coda, then one of ordinary skill in the art would have looked through each prior art reference for necessary teachings for using Coda. Each of the prior art references discloses either the same or additional features and configuration for use with Coda.

Kumar and Kistler were not before the Examiner during the prosecution of the '231 patent and have not been considered by the Office with regards to the '231 patent. Satyanarayanan is cited on the face of the '231 patent but it was not commented on by the Examiner or used in any rejection during prosecution of the '231 patent. Therefore, this combination of art describing the Coda system has not been before and has not been considered by the Office with regards to the '231 patent.

E. RELATED REEXAMINATION PROCEEDINGS

Requester is not aware of any other reexamination proceedings directly against the '231 patent, however the '231 patent is a continuation of US Pat No. 6,708,221 which is currently involved in merged reexamination proceedings with Control Nos. 90/008,131 and 90/008162. On Sept 21, 2007, the Office issued a non-final action rejecting all claims. *See* Non Final Office Action in '221 Reexam at Exhibit OTH-H. The Patent Owner has responded with arguments and amendments to attempt to overcome the rejections. *See* Patent Owner Response to Non Final Office Action in '221 Reexam at Exhibit OTH-I. As much of the prior art presented in the instant reexamination request is also at issue in the '221 reexamination, Requester has included a section below to address the Patent Owner's arguments that are relevant to the instant request and that Requester believes will be raised by the Patent Owner to attempt to overcome any rejections issued in a reexamination of the '231 patent.

In addition, three other patents assigned to the Patent Owner with similar disclosure to the '231 are undergoing reexamination. These patents and associated control numbers are:

US Patent No 6,085,192	Control No. 90/008,062
US Patent No 6,023,708	Control Nos. 90/007,421 & 90/007,933
US Patent No 7,039,679	Control No. 90/008,397

F. RELATED CO-PENDING LITIGATION REQUIRES TREATMENT WITH SPECIAL DISPATCH AND PRIORITY OVER ALL OTHER CASES

The '231 patent is presently the subject of litigation including *Research In Motion Limited vs. Visto Corporation.*, Case No. C-07-3177 (N.D. Cal. June 15, 2007) ("the Pending Litigation"). See Patent Owner's First Amended Answer to Complaint and Counterclaims filed Sept. 18, 2007 at Exhibit OTH-A.

Additionally, the '221, '192, '708, and '679 patents, from which the '231 claims priority to, are also the subject of litigation between the Patent Owner and Requester. *Visto Corp. v. Research In Motion, Ltd.*, Civil Action No. 2:06-CV-181-TJW (E.D. Tex.), filed Apr. 28, 2006.

Pursuant to 35 U.S.C. § 305, it is respectfully urged that this Request be granted and reexamination conducted not only with "**special dispatch**," but also with "**priority over all other cases**" in accordance with MPEP § 2261, due to the ongoing nature of the underlying litigation.

G. CLAIM CONSTRUCTION

Requester notes that the '231 patent, for which reexamination is requested, was asserted by Visto Corp, (hereinafter "the Patent Owner"), in the litigation *Research In Motion Limited vs. Visto Corporation.*, Case No. C-07-3177 (N.D. Cal. June 15, 2007) ("the Pending Litigation"). See Patent Owner's First Amended Answer to Complaint and Counterclaims filed Sept. 18, 2007 at Exhibit OTH-A.

Visto Corp., the assignee and Patent Owner of the '231 patent, has asserted in litigation definitions for the claim terms of the '221 patent, which the '231 patent claims priority to, as shown in the attached infringement contentions and claim construction

briefs proffered by Patent Owner in litigation. (See Exhibit OTH-B through OTH-G). For the sake of convenience, a number of Visto Corp.'s proposed or agreed to constructions applicable to the '231 patent and raised in the joint claim construction brief filed in *Visto Corp. v. Research in Motion, Ltd.*, are reproduced in the table below. Although not all terms appear in the claims of the '231 patent as written, the Patent Owner has amended many of these terms into other of its patents undergoing reexamination making all the terms relevant to possible amendments the Patent Owner may raise during a reexamination of the '231 patent.

(Visto's Proposed or Agreed Constructions from *Visto Corp. v. Research in Motion, Ltd.*, Civil Action No. 2:06-CV-181-TJW (E.D. Tex. filed September 4, 2007))

Term/Phrase	Visto's Proposed or Agreed Construction
<i>"synchronize/synchronizing"</i> ⁶	"To provide for data consistency by reconciling modifications to stored information/Providing for data consistency by reconciling modifications to stored information."
<i>"workspace data"</i>	"Data, including corresponding version information, which may include e-mail data, file data, calendar data, user data, etc. Workspace data may also include other types of data such as application programs. 'First/Second' is used as a label to denote one of one or more sets of workspace data."
<i>"workspace element/first workspace element"</i>	"A subset of workspace data such as an e-mail, file, bookmark, calendar, or applications program which may include version information."
<i>"independently modifiable copy"</i>	"A copy of a workspace element capable of being modified independent of the workspace element. The copy of the workspace element does not have to be in the same format as the workspace

⁶Italicized terms are understood to incorporate by reference the patentee Visto Corporation's own claim construction as discussed in further detail below and/or in a manner that are likely to be proffered by the patentee given its infringement contentions in the underlying litigation and its claim constructions in related litigation.

Term/Phrase	Visto's Proposed or Agreed Construction
	element.”
<i>“device/first device/second device”</i>	“An electronic device on which workspace data is stored.”
<i>“store”</i>	“A storage location for data may reside on any type of memory device. (The parties agree that this construction applies to the term “store” used in phrases “first store” and “second store.”)
<i>“server”</i>	“A computer or program that responds to commands or requests from a client. A client is a computer or program that sends commands or requests to the server.”
<i>“differences”</i>	“One or more distinctions between information or values contained in sets of data.”
<i>“normally open LAN firewall port”</i>	“A port that is typically configured to be open for packet traffic in a firewall. Ports 80 and 443 are examples of normally open ports.”
<i>“global server”</i>	“A server accessible from remote locations which stores independently modifiable copies of selected portions of workspace data.”
<i>“smart phone”</i>	“A telephone device that integrates computing capabilities and telephone capabilities.”
<i>“version information”</i>	“Information that can be used to determine a version of a workspace element.”
<i>“general synchronization module”</i>	“Software routines or code that perform the task of determining whether a workspace element and/or any independently modifiable copy thereof has (or have) been modified, based on one or more criteria.”
<i>“synchronization agent”</i>	“Software routines or code that send at least a portion of second version

Term/Phrase	Visto's Proposed or Agreed Construction
	information to a general synchronization module for purposes of synchronization.”
“ <i>synchronization start module</i> ”	“Software routines or code which initiate the synchronization process.”
“ <i>last synchronization signature</i> ”	“A record computed by the general synchronization module from which the most recent synchronization may be determined.”
“ <i>firewall</i> ”	“Software and/or hardware for protecting a network against external threats, such as hackers, coming from another network, such as the Internet.”
“ <i>HTTP port or an SSL Port</i> ”	“Any port that is used to transfer information or communicate using Hyper Text Transfer Protocol (HTTP) and any port that is used to transfer information or communicate using Secure Sockets Layer (SSL) protocol.”
“ <i>translating</i> ”	“Converting information or data in one format to information or data in another format.”
“ <i>Internet</i> ”	“A network that connects other networks, such as corporate, university, and government networks.”
“ <i>communication channel</i> ”	“A medium for transferring information. A communications channel can be a physical or wireless link.”
“ <i>communicating</i> ”	“Transmitting or transferring information.”
“ <i>means for storing first workspace data on a first device/means for storing second workspace data on a second device</i> ”	“Corresponding Structure Physical memory structure on a first device, including data storage devices 250, 350, and 720 [of the ‘221 patent]”
“ <i>synchronization means/means for</i> ”	“Corresponding Structure

Term/Phrase	Visto's Proposed or Agreed Construction
<i>synchronizing</i> "	Base system 400 (and 146) and the synchronization agent 124 ('708 patent). NOTE: base system includes the synchronization and other modules illustrated in FIG. 4"

For purposes of this Request, the Requester has and will construe all claim language from the claims asserted by the patentee in the litigation in the manner proffered by Visto Corp. Such statements by the patentee may be used by the Office to interpret claim language at issue.⁷ When the claims are interpreted in the manner proffered by Visto Corp., or even under a narrower, more reasonable interpretation of the claims, the claims are unpatentable in view of the new prior art references presented herein.

By construing the claim language in the manner proffered by Visto Corp., and/or as otherwise set forth explicitly or implicitly herein, the Requester is not admitting and/or acquiescing as to the correctness and/or reasonableness of Visto Corp.'s proffered claim construction in the litigation and/or as otherwise set forth herein. In fact, the Requester expressly challenges, and will continue to vigorously challenge, Visto Corp.'s proffered claim construction, in whole or part, as the litigation continues.

II. STATEMENT UNDER 37 C.F.R. § 1.510(B) OF EACH SUBSTANTIAL NEW QUESTION OF PATENTABILITY

This Request is based on the cited prior art documents listed above and on the accompanying Form PTO-SB/08A. Exhibit PA-SB/08A. All of these cited prior art publications constitute effective prior art as to the claims of the '231 patent under 35 U.S.C. § 102.

A. ANTICIPATION BY BROWN

Brown, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102,

⁷ See 37 C.F.R. § 1.104(c)(3): "In rejecting claims the examiner may rely upon admissions by the applicant, or the patent owner in a reexamination proceeding, as to any matter affecting patentability[.]"

claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Brown to all of the claims of the '231 patent is attached hereto at Exhibit CC-A, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

B. ANTICIPATION BY LAMB

Lamb, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Lamb to all of the claims of the '231 patent is attached hereto at Exhibit CC-B, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

C. ANTICIPATION BY LOTUS DEPLOYMENT GUIDE

Lotus Deployment Guide, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Lotus Deployment Guide to all of the claims of the '231 patent is attached hereto at Exhibit CC-C, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

D. ANTICIPATION BY LOTUS ADMIN GUIDE

Lotus Admin Guide, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Lotus Admin Guide to all of the claims of the '231 patent is attached hereto at Exhibit CC-D, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

E. OBVIOUSNESS OVER LOTUS NOTES

Brown in view of Lamb further in view of Lotus Deployment Guide further in view of Lotus Admin Guide, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, renders obvious, under 35 U.S.C. § 103, claims 1–6 of the '231 patent. Claim charts demonstrating the applicability of Brown in view of Lamb further in view of Lotus Deployment Guide further in view of Lotus Admin Guide to all of the claims of the '231 patent is attached hereto at Exhibit CC-M, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 103, and canceled pursuant to this Request.

Reasons to Combine

All four references describe with particularity the Lotus Notes system developed by IBM for synchronizing data among server and client computers. Since all of the above cited references pertain to Lotus Notes, then one of ordinary skill in the art would have looked through each prior art reference for necessary teachings for using at least Lotus Notes Revision 4. Each of the prior art references discloses either the same or additional features and configuration for use with Lotus Notes.⁸

F. ANTICIPATION BY BECKHARDT

Beckhardt, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Beckhardt to all of the claims of the '231 patent is attached hereto at Exhibit CC-E, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

⁸ In the '192 reexamination proceeding, Control No. 90/008,062, the Office has already accepted the combination of Brown, Lamb, Lotus Deployment Guide and Lotus Admin Guide under 35 U.S.C. § 103(a). See Control No. 90/008,062, Non-Final Office Action dated Sept 5, 2007 at pg. 3.

G. ANTICIPATION BY SHAHEEN

Shaheen, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Shaheen to all of the claims of the '231 patent is attached hereto at Exhibit CC-F, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

H. ANTICIPATION BY WRIGHT

Wright, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Wright to all of the claims of the '231 patent is attached hereto at Exhibit CC-G, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

I. ANTICIPATION BY GLOMOP

GloMop, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of GloMop to all of the claims of the '231 patent is attached hereto at Exhibit CC-H, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

J. ANTICIPATION BY LAMBERT

Lambert, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Lambert to all of the claims of the '231 patent is attached hereto at Exhibit CC-I, to

comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

K. ANTICIPATION BY SATYANARAYANAN

Satyanarayanan, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Satyanarayanan to all of the claims of the '231 patent is attached hereto at Exhibit CC-J, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

L. ANTICIPATION BY KUMAR

Kumar, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Kumar to all of the claims of the '231 patent is attached hereto at Exhibit CC-K, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

M. ANTICIPATION BY KISTLER

Kistler, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, fully anticipates, under 35 U.S.C. § 102, claims 1–6 of the '231 patent. A claim chart demonstrating the applicability of Kistler to all of the claims of the '231 patent is attached hereto at Exhibit CC-L, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 102, and canceled pursuant to this Request.

N. OBVIOUSNESS OVER CODA

Satyanarayanan in view of Kumar further in view of Kistler, when considered in light of the knowledge of those skilled in the art at the time of the alleged invention of the '231 patent, renders obvious, under 35 U.S.C. § 103, claims 1–6 of the '231 patent. Claim charts demonstrating the applicability of Satyanarayanan in view of Kumar further in view of Kistler to all of the claims of the '231 patent is attached hereto at Exhibit CC-N, to comply with 37 C.F.R. § 1.510(b)(2). Additionally, a SNQ raised by the present Request is demonstrated below. Accordingly claims 1–6 should be reexamined, rejected under 35 U.S.C. § 103, and canceled pursuant to this Request.

Reasons to Combine

All three references describe with particularity the Coda system developed at Carnegie Mellon University for synchronizing data between server and client computers. Since all of the above cited references pertain to Coda, then one of ordinary skill in the art would have looked through each prior art reference for necessary teachings for using Coda. Each of the prior art references discloses either the same or additional features and configuration for use with Coda.

III. CONCISE STATEMENT OF SUBSTANTIAL NEW QUESTION OF PATENTABILITY

Claims 1-6 of the '231 patent are fully anticipated under 35 U.S.C. § 102 by the several different prior art references cited herein, which were not previously considered by the Examiner during the examination of the '231 patent application or are discussed in a new light from the original prosecution of the '231 patent application. Claims 1-6 of the '231 patent are set forth in detail in the attached claim charts (Exhibit CC-A through CC-N) that compare the limitations of the claims of the '231 patent to the pertinent prior art references. As the claim charts demonstrate, claims 1-6 are unpatentable under 35 U.S.C. § 102 in view of the prior art references under any reasonable interpretation of the claims, and especially when the claims are interpreted in the manner proffered by Visto Corp. in the prior litigation.

A. Claims 1-6 are Anticipated by Brown Under 35 U.S.C. § 102

**Please see attached Exhibit CC-A
presenting claim charts for
comparison of the Brown with claims
1-6 of the '231 patent.**

Requester respectfully submits that claims 1-6 of the '231 patent are anticipated by Brown under 35 U.S.C. § 102(b). A claim chart applying Brown to these claims is submitted herewith as Exhibit CC-A.

1. Claim 1

Claim 1 of the '231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Brown.

The first element of claim 1 is "providing a first memory store including a first folder for storing first record entries in a file." Brown discloses providing a *first memory*

store including a first folder for storing first *record entries* in a file. Brown at pp. 8, 40, 571. Specifically, Brown discloses a network where each User is able to work on a client version of Lotus Notes held on their computer. Brown at pp. 8, 40, 571. This, in effect, turns their computer into a workstation. Brown at pp. 8, 40, 571. Brown further discloses that a copy of a shared database, also known as a replica, can be saved on the computer. Brown at pp. 8, 40, 571. The database is arranged in such a way that folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, the User's computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 1. Brown at pp. 8, 40, 571.

The second element of claim 1 is "providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time." Brown also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Brown at pp. 8, 40, 571. Specifically, Brown discloses that there can be multiple User workstations within the Lotus Notes network. Brown at pp. 8, 40, 571. Therefore, a *first memory store* can be one of the many computers stored on the network, and a *second memory store* can be a different User's computer on the network. Brown at pp. 8, 40, 571. Because the computers are connected over a network, they are therefore "coupled" as recited in claim 1. Brown at pp. 8, 40, 571. Also, as described above, Brown discloses that a copy of the Lotus Notes shared database can be saved on the Users' computer. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, as the database is saved on the User's computer, the User's computer includes a second folder for storing second *record entries* in a file according to claim 1. Brown at pp. 8, 40, 571.

Brown also discloses a second folder being *synchronized* with the first folder at a first memory in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Brown at pp. 8, 40, 571. Brown discloses that replica copies of the shared Lotus Notes database can be stored on the

Users' workstations. Brown at pp. 8, 40, 571. Brown further indicates that replication occurs between the computers on the Lotus Notes network. Brown at pp. 8, 40, 571. The effect of this is that it makes all copies of a database essentially identical. Brown at pp. 8, 40, 571. Thus the content of the folder on a first User's computer on the database will replicate with and be identical to the content of the same folder on a second User's computer on the network. Brown at pp. 8, 40, 571. Accordingly, this discloses that at a first moment in time, the content of "the second folder is consistent with the content of the first folder at the first moment in time." Brown at pp. 8, 40, 571.

The third element of claim 1 is "providing a global server including a third memory store having a third folder for storing third record entries in a file." Brown discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Brown at p. 8. As noted above, Brown discloses a network with multiple User workstations. Brown at p. 8. Brown further discloses a central Lotus Notes *server* that is a "main storage area for Notes databases that are shared within an organization." Brown at p. 8. Such a *server* also provides communication services to workstations and other Notes *servers*. Brown at p. 8. Such a Notes *server* can be considered a *global server* as recited in claim 1. Brown at p. 8. Furthermore, as the *server* is a storage for Notes database, it also includes a *third memory store* having a third folder for storing third *record entries* in a file as recited in claim 1. Brown at p. 8.

The fourth element of claim 1 is "receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time." Brown discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Brown at pp. 94, 582-83. Brown discloses that Users are able to make changes to the documents saved on their work stations. Brown at pp. 94, 582-83. For example, a user may be able to enter data into a new document in predefined fields. Brown at pp. 94, 582-83. Such changes disclose "receiving new first *record entries* in a file" as recited in claim 1. Brown at pp. 94, 582-83. Furthermore, Brown discloses that after documents are replicated such that they are the same across the Lotus Notes network, a User can make changes to such a document. Brown at pp. 94, 582-83. Therefore, Brown discloses that such changes occur "[a]t a second moment in time later than the first moment in time." Brown at pp.

94, 582-83.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Brown discloses electronically *synchronizing* the first folder and the third folder. Brown at pp. 571-72. Specifically, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 571-72. After changes are made to a document on a User’s workstation, the database contained on the central Notes *server* can be updated. Brown at pp. 571-72. As stated in Brown, “to update the shared database with information you created locally, you perform replication, also known as database exchange.” Brown at pp. 571-72. Additionally, the User workstation on which changes are made contains the “first folder”, and the central Notes *server* contains the “third folder”, as both have documents saved thereto. Brown at pp. 571-72. Therefore Brown discloses the recitation in claim 1 of electronically *synchronizing* the first folder and the third folder. Brown at pp. 571-72.

Additionally, as the replica on the central Notes *server* is updated with changes made on a User’s workstation, the updated changes on the central *server* can be considered a “new third *record entry*.” Brown at pp. 571-72. Furthermore, such changes to the User’s workstation are based on replication from the central *server*, thus disclosing the recitation in claim 1 that a new third *record entry* is created in a third folder based on the new first *record entries*. Brown at pp. 571-72.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Brown discloses electronically *synchronizing* the third folder and the second folder. Brown at pp. 571-72. In particular, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 571-72. After changes are updated to the central Notes *server* from a first User’s computer, the central Notes *server* then replicates to other User workstation computers in the network. Brown at pp. 571-72. Additionally, as the central Notes *server* includes the third folder, and as another User workstation has a second folder, Brown discloses *synchronizing* the third and the second folder as recited in claim 1.

Moreover, as the changes are replicated from the central Notes *server* to a User's workstation, the updated documents on the User's computer can be considered a "new second *record entry*." Brown at pp. 571-72. Furthermore, because the changes are replicated from the shared database on the central Notes *server*, the new second entry on the User's workstation is created in the second folder based on the new third *record entry* as recited in claim 1. Brown at pp. 571-72.

2. Claim 2

Claim 2 of the '231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Brown.

The first element of claim 2 is "a first memory store including a first folder for storing first record entries in a file." Brown discloses a *first memory store* including a first folder for storing first *record entries* in a file. Brown at pp. 8, 40, 571. Specifically, Brown discloses a network where each User is able to work on a client version of Lotus Notes held on their computer. Brown at pp. 8, 40, 571. This, in effect, turns their computer into a workstation. Brown at pp. 8, 40, 571. Brown further discloses that a copy of a shared database can be saved on the computer, also known as a replica. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to

hold electronic documents. Brown at pp. 8, 40, 571. Therefore, the User's computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 2. Brown at pp. 8, 40, 571.

The second element of claim 2 is "a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time." Brown also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Brown at pp. 8, 40, 569, 571. Specifically, Brown discloses that there can be multiple User workstations within the Lotus Notes network. Brown at pp. 8, 40, 569, 571. Therefore, a *first memory store* can be one of the many computers stored on the network, and a *second memory store* can be a different User's computer on the network. Brown at pp. 8, 40, 569, 571. As the computers are connected over a network, they are therefore "coupled" as recited in claim 2. Brown at pp. 8, 40, 569, 571. Also, as described above, Brown discloses that a copy of the Lotus Notes shared database can be saved on the Users' computer. Brown at pp. 8, 40, 569, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 569, 571. Therefore, as the database is saved on the User's computer, the User's computer includes a second folder for storing second *record entries* in a file according to claim 2. Brown at pp. 8, 40, 569, 571.

Brown also discloses a second folder being *synchronized* with the first folder at a first memory in time such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Brown at pp. 8, 40, 569, 571. Brown discloses that replica copies of the shared Lotus Notes database can be stored on the Users' workstations. Brown at pp. 8, 40, 569, 571. Brown further indicates that replication occurs between the computers on the Lotus Notes network. Brown at pp. 8, 40, 569, 571. The effect of this is that it makes all copies of a database essentially identical. Brown at pp. 8, 40, 569, 571. Thus the content of the folder on a first User's computer on the database will replicate with and be identical to the content of the same folder on a second User's computer on the network. Brown at pp. 8, 40, 569, 571.

Accordingly, this discloses that at a first moment in time, the content of “the second folder is consistent with the content of the first folder at the first moment in time.” Brown at pp. 8, 40, 569, 571.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Brown discloses an interface for receiving new *record entries* in a file at the *first memory store*. Brown at pp. 94, 582-83. Brown discloses that Users are able to make changes to the documents saved on their work stations. Brown at pp. 94, 582-83. For example, a user may be able to enter data into a new document in predefined fields. Brown at pp. 94, 582-83. Such changes disclose receiving new *record entries* in a file at the *first memory store*. Brown at pp. 94, 582-83. Furthermore, the User’s workstation may be considered an “interface.”

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Brown discloses a general synchronization module for electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84. Specifically, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 40 571-72, 583-84. After changes are made to a document on a User’s workstation, the database contained on the central Notes *server* can be updated. Brown at pp. 40 571-72, 583-84. As stated in Brown, “to update the shared database with information you created locally, you perform replication, also known as database exchange.” Brown at pp. 40 571-72, 583-84. Additionally, the User workstation on which changes are made contains the “first folder”, and the central Notes *server* contains the “third folder”, as both have documents saved thereto. Brown at pp. 40 571-72, 583-84. Therefore Brown discloses the recitation in claim 2 of electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84.

Moreover, as the replica on the central Notes *server* is updated with changes made on a User’s workstation, the updated changes on the central *server* can be considered a “new third *record entry*.” Brown at pp. 40 571-72, 583-84. Furthermore, such changes to the User’s workstation are based on replication from the central *server*, thus disclosing

the recitation in claim 2 that a new third *record entry* is created in a third folder based on the new first *record entries*. Brown at pp. 40 571-72, 583-84.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Brown discloses electronically *synchronizing* the third folder and the second folder. Brown at pp. 571-72, 583-84. In particular, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 571-72, 583-84. After changes are updated to the central Notes *server* from a first User’s computer, the central Notes *server* then replicates to other User workstation computers in the network. Brown at pp. 571-72, 583-84. Additionally, as the central Notes *server* includes the third folder, and as another User workstation has a second folder, Brown discloses *synchronizing* the third and the second folder as recited in claim 2. Brown at pp. 571-72, 583-84.

Moreover, as the changes are replicated from the central Notes *server* to a User’s workstation, the updated documents on the User’s computer can be considered a “new second *record entry*.” Brown at pp. 571-72, 583-84. Furthermore, because the changes are replicated from the shared database on the central Notes *server*, the new second entry on the User’s workstation is created in the second folder based on the new third *record entry* as recited in claim 2. Brown at pp. 571-72, 583-84.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Brown.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Brown discloses a *first memory store* including a first folder for storing first *record entries* in a file. Brown at pp. 8, 40, 571. Specifically, Brown discloses a network where each User is able to work on a client version of Lotus Notes held on their computer. Brown at pp. 8, 40, 571. This, in effect, turns their computer into a workstation. Brown at pp. 8, 40, 571. Brown further discloses that a copy of a shared database can be saved on the computer, also known as a replica. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, the User’s computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 3. Brown at pp. 8, 40, 571.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Brown also discloses a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Brown at pp. 8, 40, 569, 571. Specifically, Brown discloses that there can be multiple User workstations within the Lotus Notes network. Brown at pp. 8, 40, 569, 571. Therefore, a first memory means can be one of the many computers on the network, and a second memory means can be a different User’s computer on the network. Brown at pp. 8, 40, 569, 571. As the

computers are connected over a network, they are therefore “coupled” as recited in claim 3. Brown at pp. 8, 40, 569, 571. Also, as described above, Brown discloses that a copy of the Lotus Notes shared database can be saved on the Users’ computer. Brown at pp. 8, 40, 569, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 569, 571. Therefore, as the database is saved on the User’s computer, Brown discloses that a User’s computer includes a second memory means including a second folder for storing second *record entries* in a file according to claim 3. Brown at pp. 8, 40, 569, 571.

Brown also discloses a second folder being *synchronized* with the first folder at a first memory in time such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Brown at pp. 8, 40, 569, 571. Brown discloses that replica copies of the shared Lotus Notes database can be stored on the Users’ workstations. Brown at pp. 8, 40, 569, 571. Brown further indicates that replication occurs between the computers on the Lotus Notes network. Brown at pp. 8, 40, 569, 571. The effect of this is that it makes all copies of a database essentially identical. Brown at pp. 8, 40, 569, 571. Thus the content of the folder on a first User’s computer on the database will replicate with and be identical to the content of the same folder on a second User’s computer on the network. Brown at pp. 8, 40, 569, 571. Accordingly, this discloses that at a first moment in time, the content of “the second folder is consistent with the content of the first folder at the first moment in time.” Brown at pp. 8, 40, 569, 571.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Brown discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Brown at p. 8. As noted above, Brown discloses a network with multiple User workstations. Brown at p. 8. Brown further discloses a central Lotus Notes *server* that is a “main storage area for Notes databases that are shared within an organization.” Brown at p. 8. Such a *server* also provides communication services to workstations and other Notes *servers*. Brown at p. 8. Such a Notes *server* can be considered a *global server* as recited in claim 3. Brown at p. 8. Furthermore, as the *server* is storage for Notes database, it also includes a *third memory store* having a third

folder for storing third *record entries* in a file as recited in claim 3. Brown at p. 8.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Brown discloses a means for receiving new *record entries* in a file at the *first memory store*. Brown at pp. 94, 582-83. Brown discloses that Users are able to make changes to the documents saved on their work stations. Brown at pp. 94, 582-83. For example, a user may be able to enter data into a new document in predefined fields. Brown at pp. 94, 582-83. Such changes disclose receiving new *record entries* in a file at the *first memory store*. Brown at pp. 94, 582-83.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Brown discloses a means for electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84. Specifically, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 40 571-72, 583-84. After changes are made to a document on a User’s workstation, the database contained on the central Notes *server* can be updated. Brown at pp. 40 571-72, 583-84. As stated in Brown, “to update the shared database with information you created locally, you perform replication, also known as database exchange.” Brown at pp. 40 571-72, 583-84. Additionally, the User workstation on which changes are made contains the “first folder”, and the central Notes *server* contains the “third folder”, as both have documents saved thereto. Brown at pp. 40 571-72, 583-84. Therefore Brown discloses the recitation in claim 3 of electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84.

Moreover, as the replica on the central Notes *server* is updated with changes made on a User’s workstation, the updated changes on the central *server* can be considered a “new third *record entry*.” Brown at pp. 40 571-72, 583-84. Furthermore, such changes to the User’s workstation are based on replication from the central *server*, thus disclosing the recitation in claim 3 that a new third *record entry* is created in a third folder based on the new first *record entries*. Brown at pp. 40 571-72, 583-84.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second

folder based on the new third record entry.” Brown discloses electronically *synchronizing* the third folder and the second folder. Brown at pp. 571-72, 583-84. In particular, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 571-72, 583-84. After changes are updated to the central Notes *server* from a first User’s computer, the central Notes *server* then replicates to other User workstation computers in the network. Brown at pp. 571-72, 583-84. Additionally, as the central Notes *server* includes the third folder, and as another User workstation has a second folder, Brown discloses *synchronizing* the third and the second folder as recited in claim 3. Brown at pp. 571-72, 583-84.

Furthermore, as the changes are replicated from the central Notes *server* to a User’s workstation, the updated documents on the User’s computer can be considered a “new second *record entry*.” Brown at pp. 571-72, 583-84. Furthermore, because the changes are replicated from the shared database on the central Notes *server*, the new second entry on the User’s workstation is created in the second folder based on the new third *record entry* as recited in claim 3. Brown at pp. 571-72, 583-84.

4. Claim 4

Claim 4 of the ‘231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Brown.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Brown discloses a *first memory store* including a first folder for storing first *record entries* in a file. Brown at pp. 8, 40, 571. Specifically, Brown discloses a network where each User is able to work on a client version of Lotus Notes held on their computer. Brown at pp. 8, 40, 571. This, in effect, turns their computer into a workstation. Brown at pp. 8, 40, 571. Brown further discloses that a copy of a shared database can be saved on the computer, also known as a replica. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, the User’s computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 4. Brown at pp. 8, 40, 571.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Brown also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Brown at pp. 8, 40, 571. Specifically, Brown discloses that there can be multiple User workstations within the Lotus Notes network. Brown at pp. 8, 40, 571. at pp. 8 Therefore, a *first memory store* can be one of the many computers stored on the network, and a *second memory store* can be a different User’s computer on the network. Brown at pp. 8, 40, 571. As the computers are connected over a network, they are therefore “coupled” as recited in claim 4. Brown at pp. 8, 40, 571. Also, as described above, Brown discloses that a copy of the Lotus Notes shared database can be saved on the Users’ computer. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, as the database is saved on the User’s

computer, the User's computer includes a second folder for storing second *record entries* in a file according to claim 4. Brown at pp. 8, 40, 571.

Brown also discloses a second folder being *synchronized* with the first folder at a first memory in time such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Brown at pp. 8, 40, 571. Brown discloses that replica copies of the shared Lotus Notes database can be stored on the Users' workstations. Brown at pp. 8, 40, 571. Brown further indicates that replication occurs between the computers on the Lotus Notes network. Brown at pp. 8, 40, 571. The effect of this is that it makes all copies of a database essentially identical. Brown at pp. 8, 40, 571. Thus the content of the folder on a first User's computer on the database will replicate with and be identical to the content of the same folder on a second User's computer on the network. Brown at pp. 8, 40, 571. Accordingly, this discloses that at a first moment in time, the content of "the second folder is consistent with the content of the first folder at the first moment in time." Brown at pp. 8, 40, 571.

The third element of claim 4 is "providing a global server including a third memory store having a third folder for storing third record entries in a file." Brown discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Brown at p. 8. As noted above, Brown discloses a network with multiple User workstations. Brown at p. 8. Brown further discloses a central Lotus Notes *server* that is a "main storage area for Notes databases that are shared within an organization." Brown at p. 8. Such a *server* also provides communication services to workstations and other Notes *servers*. Brown at p. 8. Such a Notes *server* can be considered a *global server* as recited in claim 4. Brown at p. 8. Furthermore, as the *server* is storage for Notes database, it also includes a *third memory store* having a third folder for storing third *record entries* in a file as recited in claim 4. Brown at p. 8.

The fourth element of claim 4 is "receiving changes at the first memory store." Brown discloses receiving changes the *first memory store*. Brown at pp. 94, 582-83. Brown discloses that Users are able to make changes to the documents saved on their work stations. Brown at pp. 94, 582-83. For example, a user may be able to enter data into a new document in predefined fields. Brown at pp. 94, 582-83. Such changes

disclose receiving new *record entries* in a file at the *first memory store*. Brown at pp. 94, 582-83.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Brown discloses electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84. Specifically, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 40 571-72, 583-84. After changes are made to a document on a User’s workstation, the database contained on the central Notes *server* can be updated. Brown at pp. 40 571-72, 583-84. As stated in Brown, “to update the shared database with information you created locally, you perform replication, also known as database exchange.” Brown at pp. 40 571-72, 583-84. Additionally, the User workstation on which changes are made contains the “first folder”, and the central Notes *server* contains the “third folder”, as both have documents saved thereto. Brown at pp. 40 571-72, 583-84. Therefore Brown discloses the recitation in claim 4 of electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84.

Moreover, as the replica on the central Notes *server* is updated with changes made on a User’s workstation, the updated changes on the central *server* can be considered a “third *record entry*.” Brown at pp. 40 571-72, 583-84. Furthermore, such changes to the User’s workstation are based on replication from the central *server*, thus disclosing the recitation in claim 4 that the content of the changes creates a third *record entry*. Brown at pp. 40 571-72, 583-84.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Brown discloses electronically *synchronizing* the third folder and the second folder. Brown at pp. 571-72. In particular, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 571-72. After changes are updated to the central Notes *server* from a first User’s computer, the central Notes *server* then replicates to other User workstation computers in the network.

Brown at pp. 571-72. Additionally, as the central Notes *server* includes the third folder, and as another User workstation has a second folder, Brown discloses *synchronizing* the third and the second folder as recited in claim 4.

Moreover, as the changes are replicated from the central Notes *server* to a User's workstation, the updated documents on the User's computer can be considered a "new second *record entry*." Brown at pp. 571-72. Furthermore, because the changes are replicated from the shared database on the central Notes *server*, the new second entry on the User's workstation is created in the second folder based on the new third *record entry* as recited in claim 4. Brown at pp. 571-72.

5. Claim 5

Claim 5 of the '231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Brown.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Brown discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Brown at pp. 8, 40, 571.

Specifically, Brown discloses a network where each User is able to work on a client version of Lotus Notes held on their computer. Brown at pp. 8, 40, 571. This, in effect, turns their computer into a workstation. Brown at pp. 8, 40, 571. Brown further discloses that a copy of a shared database can be saved on the computer, also known as a replica. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, the User’s computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 5. Brown at pp. 8, 40, 571.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Brown also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Brown at pp. 8, 40, 571. Specifically, Brown discloses that there can be multiple User workstations within the Lotus Notes network. Brown at pp. 8, 40, 571. Brown at pp. 8, 40, 571. Therefore, a *first memory store* can be one of the many computers stored on the network, and a *second memory store* can be a different User’s computer on the network. Brown at pp. 8, 40, 571. As the computers are connected over a network, they are therefore “coupled” as recited in claim 5. Brown at pp. 8, 40, 571. Also, as described above, Brown discloses that a copy of the Lotus Notes shared database can be saved on the Users’ computer. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, as the database is saved on the User’s computer, the User’s computer includes a second folder for storing second *record entries* in a file according to claim 5. Brown at pp. 8, 40, 571.

Brown also discloses a second folder being *synchronized* with the first folder at a first memory in time such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Brown at pp. 8, 40, 571. Brown discloses that replica copies of the shared Lotus Notes database can be stored on the Users' workstations. Brown at pp. 8, 40, 571. Brown further indicates that replication occurs between the computers on the Lotus Notes network. Brown at pp. 8, 40, 571. The effect of this is that it makes all copies of a database essentially identical. Brown at pp. 8, 40, 571. Thus the content of the folder on a first User's computer on the database will replicate with and be identical to the content of the same folder on a second User's computer on the network. Brown at pp. 8, 40, 571. Accordingly, this discloses that at a first moment in time, the content of "the second folder is consistent with the content of the first folder at the first moment in time." Brown at pp. 8, 40, 571.

The third element of claim 5 is "providing a global server including a third memory store having a third folder for storing third record entries in a file." Brown discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Brown at p. 8. As noted above, Brown discloses a network with multiple User workstations. Brown at p. 8. Brown further discloses a central Lotus Notes *server* that is a "main storage area for Notes databases that are shared within an organization." Brown at p. 8. Such a *server* also provides communication services to workstations and other Notes *servers*. Brown at p. 8. Such a Notes *server* can be considered a *global server* as recited in claim 5. Brown at p. 8. Furthermore, as the *server* is storage for Notes database, it also includes a *third memory store* having a third folder for storing third *record entries* in a file as recited in claim 5. Brown at p. 8.

The fourth element of claim 5 is "an interface for receiving changes at the first memory store." Brown discloses an interface for receiving changes at the *first memory store*. Brown at pp. 94, 582-83. Brown discloses that Users are able to make changes to the documents saved on their work stations. Brown at pp. 94, 582-83. For example, a user may be able to enter data into a new document in predefined fields. Brown at pp. 94, 582-83. Such changes disclose changes at the *first memory store*. Brown at pp. 94, 582-83. Furthermore, the User's workstation may be considered an "interface."

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Brown discloses a general synchronization module for electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84. Specifically, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 40 571-72, 583-84. After changes are made to a document on a User’s workstation, the database contained on the central Notes *server* can be updated. Brown at pp. 40 571-72, 583-84. As stated in Brown, “to update the shared database with information you created locally, you perform replication, also known as database exchange.” Brown at pp. 40 571-72, 583-84. Additionally, the User workstation on which changes are made contains the “first folder”, and the central Notes *server* contains the “third folder”, as both have documents saved thereto. Brown at pp. 40 571-72, 583-84. Therefore Brown discloses the recitation in claim 5 of electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84.

Moreover, as the replica on the central Notes *server* is updated with changes made on a User’s workstation, the updated changes on the central *server* can be considered a “new third *record entry*.” Brown at pp. 40 571-72, 583-84. Furthermore, such changes to the User’s workstation are based on replication from the central *server*, thus disclosing the recitation in claim 5 that a new third *record entry* is created in a third folder based on the new first *record entries*. Brown at pp. 40 571-72, 583-84.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Brown discloses electronically *synchronizing* the third folder and the second folder. Brown at pp. 571-72, 583-84. In particular, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 571-72, 583-84. After changes are updated to the central Notes *server* from a first User’s computer, the central Notes *server* then replicates to other User workstation computers in the network. Brown at pp. 571-72, 583-84. Additionally, as the central Notes *server* includes the third folder, and as another User workstation has a second folder, Brown discloses *synchronizing* the third and the

second folder as recited in claim 5. Brown at pp. 571-72, 583-84.

Moreover, as the changes are replicated from the central Notes *server* to a User's workstation, the updated documents on the User's computer can be considered a "new second *record entry*." Brown at pp. 571-72, 583-84. Furthermore, because the changes are replicated from the shared database on the central Notes *server*, the new second entry on the User's workstation is created in the second folder based on the new third *record entry* as recited in claim 5. Brown at pp. 571-72, 583-84.

6. Claim 6

Claim 6 of the '231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Brown.

The first element of claim 6 is "first memory means including a first folder for storing first record entries in a file." Brown discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Brown at pp. 8, 40, 571.

Specifically, Brown discloses a network where each User is able to work on a client version of Lotus Notes held on their computer. Brown at pp. 8, 40, 571. This, in

effect, turns their computer into a workstation. Brown at pp. 8, 40, 571. Brown further discloses that a copy of a shared database can be saved on the computer, also known as a replica. Brown at pp. 8, 40, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 571. Therefore, the User's computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 6. Brown at pp. 8, 40, 571.

The second element of claim 6 is "second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time." Brown also discloses a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Brown at pp. 8, 40, 569, 571. Specifically, Brown discloses that there can be multiple User workstations within the Lotus Notes network. Brown at pp. 8, 40, 569, 571. Therefore, a first memory means can be one of the many computers on the network, and a second memory means can be a different User's computer on the network. Brown at pp. 8, 40, 569, 571. As the computers are connected over a network, they are therefore "coupled" as recited in claim 6. Brown at pp. 8, 40, 569, 571. Also, as described above, Brown discloses that a copy of the Lotus Notes shared database can be saved on the Users' computer. Brown at pp. 8, 40, 569, 571. The database is arranged in a way where folders can be organized to hold electronic documents. Brown at pp. 8, 40, 569, 571. Therefore, as the database is saved on the User's computer, Brown discloses that a User's computer includes a second memory means including a second folder for storing second *record entries* in a file according to claim 6. Brown at pp. 8, 40, 569, 571.

Brown also discloses a second folder being *synchronized* with the first folder at a first memory in time such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Brown at pp. 8, 40, 569, 571. Brown discloses that replica copies of the shared Lotus Notes database can be stored on the Users' workstations. Brown at pp. 8, 40, 569, 571. Brown further indicates that replication occurs between the computers on the Lotus Notes network. Brown at pp. 8,

40, 569, 571. The effect of this is that it makes all copies of a database essentially identical. Brown at pp. 8, 40, 569, 571. Thus the content of the folder on a first User's computer on the database will replicate with and be identical to the content of the same folder on a second User's computer on the network. Brown at pp. 8, 40, 569, 571. Accordingly, this discloses that at a first moment in time, the content of "the second folder is consistent with the content of the first folder at the first moment in time." Brown at pp. 8, 40, 569, 571.

The third element of claim 6 is "providing a global server including a third memory means having a third folder for storing third record entries in a file." Brown discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Brown at p. 8. As noted above, Brown discloses a network with multiple User workstations. Brown at p. 8. Brown further discloses a central Lotus Notes *server* that is a "main storage area for Notes databases that are shared within an organization." Brown at p. 8. Such a *server* also provides communication services to workstations and other Notes *servers*. Brown at p. 8. Such a Notes *server* can be considered a *global server* as recited in claim 6. Brown at p. 8. Furthermore, as the *server* is storage for Notes database, it also includes a *third memory store* having a third folder for storing third *record entries* in a file as recited in claim 6. Brown at p. 8.

The fourth element of claim 6 is "means for receiving changes at the first memory means." Brown discloses receiving changes to the *first memory store*. Brown at pp. 94, 582-83. Brown discloses that Users are able to make changes to the documents saved on their work stations. Brown at pp. 94, 582-83. For example, a user may be able to enter data into a new document in predefined fields. Brown at pp. 94, 582-83. Such changes disclose receiving new *record entries* in a file at the *first memory store*. Brown at pp. 94, 582-83.

The fifth element of claim 6 is "means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry." Brown discloses a means for electronically *synchronizing* the first folder and the third folder. Brown at pp. 40 571-72, 583-84. Specifically, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp.

40 571-72, 583-84. After changes are made to a document on a User's workstation, the database contained on the central Notes *server* can be updated. Brown at pp. 40 571-72, 583-84. As stated in Brown, "to update the shared database with information you created locally, you perform replication, also known as database exchange." Brown at pp. 40 571-72, 583-84. Additionally, the User workstation on which changes are made contains the "first folder", and the central Notes *server* contains the "third folder", as both have documents saved thereto. Brown at pp. 40 571-72, 583-84. Therefore Brown discloses the recitation in claim 6 of electronically *synchronizing* the first folder and the third folder." Brown at pp. 40 571-72, 583-84.

Moreover, as the replica on the central Notes *server* is updated with changes made on a User's workstation, the updated changes on the central *server* can be considered a new third *record entry*. Brown at pp. 40 571-72, 583-84. Furthermore, such changes to the User's workstation are based on replication from the central *server*, thus disclosing the recitation in claim 6 that a new third *record entry* is created in a third folder based on the new first *record entries*. Brown at pp. 40 571-72, 583-84.

The last element of claim 6 is "means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry." Brown discloses electronically *synchronizing* the third folder and the second folder. Brown at pp. 571-72, 583-84. In particular, Brown discloses that a copy of a shared database is stored on workstations as well as a central *server*. Brown at pp. 571-72, 583-84. After changes are updated to the central Notes *server* from a first User's computer, the central Notes *server* then replicates to other User workstation computers in the network. Brown at pp. 571-72, 583-84. Additionally, as the central Notes *server* includes the third folder, and as another User workstation has a second folder, Brown discloses *synchronizing* the third and the second folder as recited in claim 6. Brown at pp. 571-72, 583-84.

Moreover, as the changes are replicated from the central Notes *server* to a User's workstation, the updated documents on the User's computer can be considered a "new second *record entry*." Brown at pp. 571-72, 583-84. Furthermore, because the changes are replicated from the shared database on the central Notes *server*, the new second entry

on the User's workstation is created in the second folder based on the new third *record entry* as recited in claim 6. Brown at pp. 571-72, 583-84.

B. Claims 1-6 are Anticipated by Lamb Under 35 U.S.C. § 102

**Please see attached Exhibit CC-B
presenting claim charts for
comparison of the Lamb with claims
1-6 of the '231 patent.**

Requester respectfully submits that claims 1-6 of the '231 patent are anticipated by Lamb under 35 U.S.C. § 102(b). A claim chart applying Lamb to these claims is submitted herewith as Exhibit CC-B.

1. Claim 1

Claim 1 of the '231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

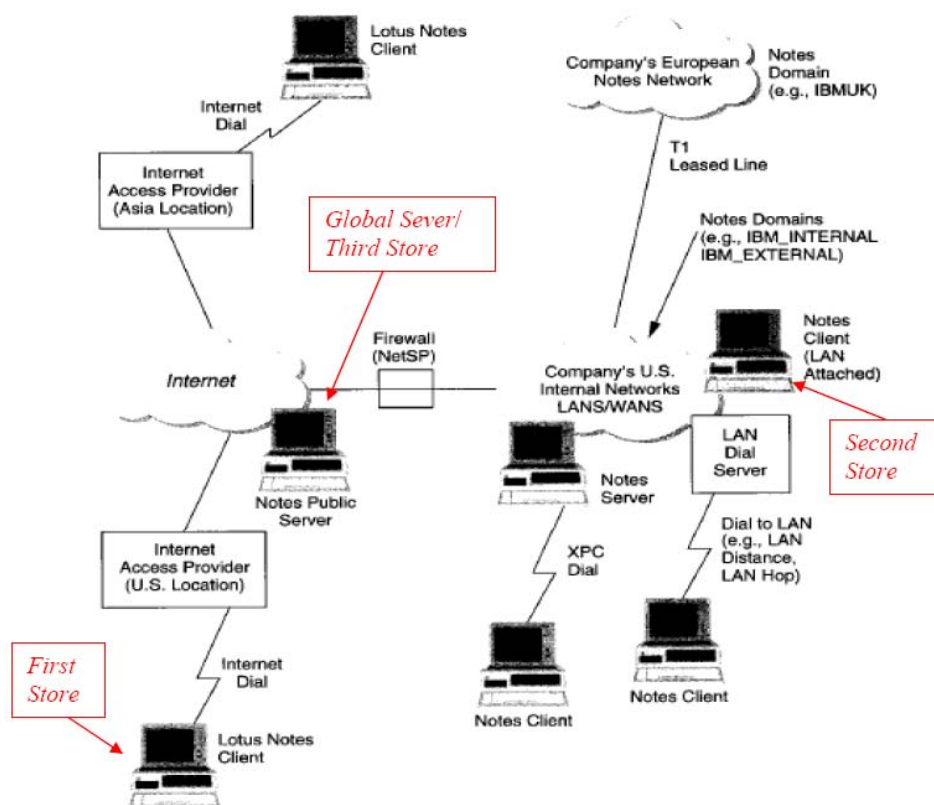
electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Lamb.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Lamb discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lamb at pp. 7, 178 at Fig. 2.1.

Specifically, Lamb discloses a “hub and spoke” concept having a Lotus Notes public server connected in a network with various Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Therefore, a *first memory store* can be one of the Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Lamb also discloses that a client can include a folder containing a Lotus Notes database with modifiable fields. Lamb at 28. Thus, the client computers of Lamb include a first folder for storing first *record entries*.



Lamb Fig. 2.1

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the

content of the first folder at the first moment in time.” Lamb also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. As discussed above, Lamb discloses a network with a hub and spoke architecture, with the Lotus Notes *server* connected with client computers, and replicating to these client computers. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. The client computers can be mobile computers such as laptops or notebooks, as well as PDAs. Lamb at pp. 9, 27-28, 178, 196. One laptop or PDA can be a *first memory store* in the network, and a different laptop or PDA can be a *second memory store*. Lamb at pp. 9, 27-28, 178, 196. Because they are connected in a network, they are therefore “coupled” as recited in claim 4. Lamb at pp. 9, 27-28, 178, 196 and Fig 2.1.

These laptops or PDAs can include a folder containing a database for Lotus Notes as well as Notes applications. Lamb at pp. 9, 27-28, 178, 196. Thus, a second laptop or PDA in the network includes a folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196. Furthermore, Lamb discloses that Lotus Notes documents can be saved in the database on the *server* and client computers in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Thus, the documents are “files” according to claim 1, and the updatable fields are *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

Lamb also discloses the second folder being *synchronized* with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Lamb at pp. 9, 27-28, 178, 196. In the hub and spoke network disclosed by Lamb, the hub, also known as the Lotus Notes *server*, replicates to the spokes, also known as the Lotus Notes client computers. Lamb also discloses replicating only the modified fields of a larger database document in order to reduce the amount of data transferred. Lamb at pp. 9, 27-28, 178, 196. As stated in Lamb, with “the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*. Lamb at pp. 9, 27-28, 178, 196. Then the hub *server* will replicate to all other spoke *servers*.” Lamb at pp. 9, 27-28, 178, 196. Therefore, a first client computer can replicate to a central hub *server*, which then can replicate that data to a second client computer. Lamb at pp. 9, 27-28, 178, 196. Thus, at a first moment in

time, Lamb discloses that the content of the documents stored in the multiple client computers will be *synchronized*. Lamb at pp. 9, 27-28, 178, 196.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lamb discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lamb at pp. 27, 178. As discussed above, Lamb discloses a hub and spoke network architecture. Lamb at pp. 27, 178. The hub acts as a central Lotus Notes *server* and thus can be considered a *global server* including a *third memory store* as recited in claim 1. Lamb at pp. 27, 178.

Additionally, the central hub *server* can have saved thereto “master copies of each database.” Lamb at pp. 27, 178. Furthermore, as noted above, Lotus Notes documents can be saved in the database on the central *server* in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Therefore, the central hub has a third folder for storing third *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Lamb discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Lamb at pp. 5-6, 28. In particular, Lamb discloses that a User can do work on a Lotus Notes client, independent of the Notes *server*. Lamb at pp. 5-6, 28. Additionally, Release 4 of Notes enables “updated fields” across the network. Lamb at pp. 5-6, 28. Thus, changes can be made to “fields” in the Lotus Notes documents. Lamb at pp. 5-6, 28. Changes to such “fields” are “receiving new first *record entries*” according to claim 1. Lamb at pp. 5-6, 28. Furthermore, because the changes are made relative to original documents that are replicated to client computers across the network, they are at a “second moment in time later than the first moment in time.” Lamb at pp. 5-6, 28.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Lamb discloses electronically *synchronizing* the first folder

and the third folder. Lamb at pp. 9, 28. Specifically, Lamb discloses that “[w]ith the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*...[t]hen the hub *server* will replicate to all other *servers*.” Lamb at pp. 9, 28. Therefore, Lamb discloses that changes made to documents on the client spoke *servers* are replicated to the central hub *servers*. Lamb at pp. 9, 28. Because the client *server* on which changes are made contains the “first folder,” and the central hub *server* contains the “third folder,” therefore in Lamb, a first folder on a client *server* is *synchronized* with the third folder on the central hub *server* as recited in claim 1. Lamb at pp. 9, 28.

Lamb discloses that synchronization is such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lamb at pp. 9, 28. Release 4 of Lotus Notes enables updated “fields” in documents across the network. Such fields are *record entries*. Therefore, when the fields are replicated to the central hub, this discloses that a new third *record entry* is created and based on the first *record entry* from the client *server* as recited in claim 1.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lamb discloses electronically *synchronizing* the third folder and the second folder. Lamb at pp. 9, 28. In particular, Lamb discloses that after a client spoke *server* replicates to the central hub *server*, “[t]hen the hub *server* will replicate to all other spoke *servers*.” Therefore, after replicating to the central *server*, the changes are then replicated to other client computers. Thus, electronically *synchronizing* the third and the second folder as recited in claim 1 is disclosed.

Lamb also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. Furthermore, Lamb discloses that Release 4 of Lotus Notes enables updated fields across the network. As discussed above, such fields in the Lotus Notes documents can be considered *record entries* in a file. Because such fields are replicated from the hub central *server* to client *servers*, such as laptops or PDAs, these changes create a “new second entry” as recited in claim 1. Such new *record entry* on the client *server* is replicated from the central hub *server* and so is based on the new third *record entry* also as recited in claim 1.

2. Claim 2

Claim 2 of the '231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

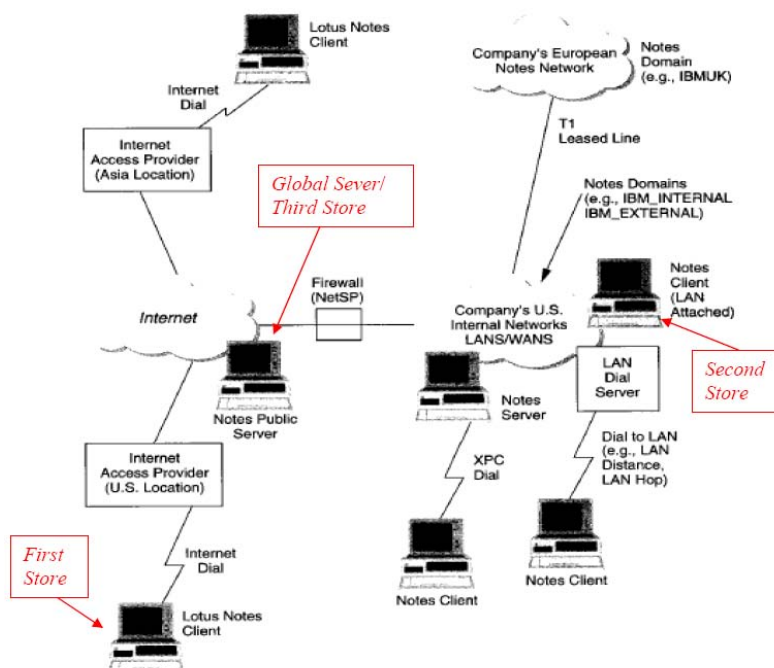
an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Lamb.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Lamb discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lamb at pp. 7, 178 at Fig. 2.1. Specifically, Lamb discloses a “hub and spoke” concept having a Lotus Notes public *server* connected in a network with various Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Therefore, a *first memory store* can be one of the Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Lamb also discloses that a client can include a folder containing a Lotus Notes database with modifiable fields. Lamb at 28. Thus, the client computers of Lamb include a first folder for storing first *record entries*.



Lamb Fig. 2.1

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lamb also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. As discussed above, Lamb discloses a network with a hub and spoke architecture, with the Lotus Notes *server* connected with client computers, and replicating to these client computers. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. The client computers can be mobile computers such as laptops or notebooks, as well as PDAs. Lamb at pp. 9, 27-28, 178, 196. One laptop or PDA can be a *first memory store* in the network, and a different laptop or PDA can be a *second memory store*. Lamb at pp. 9, 27-28, 178, 196. Because they are connected in a network, they are therefore “coupled” as recited in claim 4. Lamb at pp. 9, 27-28, 178, 196 and Fig 2.1.

These laptops or PDAs can include a folder containing a database for Lotus Notes as well as Notes applications. Lamb at pp. 9, 27-28, 178, 196. Thus, a second laptop or

PDA in the network includes a folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196. Furthermore, Lamb discloses that Lotus Notes documents can be saved in the database on the *server* and client computers in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Thus, the documents are “files” according to claim 2, and the updatable fields are *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

Lamb also discloses the second folder being *synchronized* with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Lamb at pp. 9, 27-28, 178, 196. In the hub and spoke network disclosed by Lamb, the hub, also known as the Lotus Notes *server*, replicates to the spokes, also known as the Lotus Notes client computers. Lamb also discloses replicating only the modified fields of a larger database document in order to reduce the amount of data transferred. Lamb at pp. 9, 27-28, 178, 196. As stated in Lamb, with “the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*. Lamb at pp. 9, 27-28, 178, 196. Then the hub *server* will replicate to all other spoke *servers*.” Lamb at pp. 9, 27-28, 178, 196. Therefore, a first client computer can replicate to a central hub *server*, which then can replicate that data to a second client computer. Lamb at pp. 9, 27-28, 178, 196. Thus, at a first moment in time, Lamb discloses that the content of the documents stored in the multiple client computers will be *synchronized*. Lamb at pp. 9, 27-28, 178, 196.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Lamb discloses an interface for receiving new *record entries* in a file at the *first memory store*. Lamb at pp. 5-6, 28. In particular, Lamb discloses that a User can do work on a Lotus Notes client, independent of the Notes *server*. Lamb at pp. 5-6, 28. A *server*, such as a laptop or PDA on which Users work, according to Lamb, can be considered “an interface” as recited in claim 2. Additionally, Release 4 of Notes enables “updated fields” across the network. Lamb at pp. 5-6, 28. Thus, changes can be made to “fields” in the Lotus Notes documents. Lamb at pp. 5-6, 28. Changes to such “fields” are receiving new first *record entries* in a file according to claim 2. Lamb at pp. 5-6, 28.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Lamb discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Lamb at pp. 9, 27-28, 178. Specifically, Lamb discloses that “[w]ith the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*...[t]hen the hub *server* will replicate to all other *servers*.” Lamb at pp. 9, 27-28, 178. Therefore, Lamb discloses that changes made to documents on the client spoke *servers* are replicated to the central hub *servers*. Lamb at pp. 9, 27-28, 178. Because the client *server* on which changes are made contains the “first folder,” and the central hub *server* contains the “third folder,” therefore in Lamb, a first folder on a client *server* is *synchronized* with the third folder on the central hub *server* as recited in claim 2. Lamb at pp. 9, 27-28, 178. Additionally, the “general synchronization module” can be either the central *server* or *servers*, laptops, or PDAs on which the Users work. Lamb at pp. 9, 27-28, 178.

Lamb discloses that synchronization is such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lamb at pp. 9, 27-28, 178. Release 4 of Lotus Notes enables updated “fields” in documents across the network. Lamb at pp. 9, 27-28, 178. Such fields are *record entries*. Lamb at pp. 9, 27-28, 178. Therefore, when the fields are replicated to the central hub, this discloses that a new third *record entry* is created and based on the first *record entry* from the client *server* as recited in claim 2. Lamb at pp. 9, 27-28, 178.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Lamb discloses an electronically *synchronizing* agent for electronically *synchronizing* the third folder and the second folder. Lamb at pp. 9, 28. In particular, Lamb discloses that after a client spoke *server* replicates to the central hub *server*, “[t]hen the hub *server* will replicate to all other spoke *servers*.” Therefore, after replicating to the central *server*, the changes are then replicated to other client computers. Thus, electronically *synchronizing* the third and the second folder as recited in claim 2 is disclosed. Furthermore, the “synchronization

agent” can be either the central *server* or *servers* on which the Users work. Lamb at pp. 9, 28.

Lamb also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. Lamb at pp. 9, 28. Furthermore, Lamb discloses that Release 4 of Lotus Notes enables updated fields across the network. Lamb at pp. 9, 28. As discussed above, such fields in the Lotus Notes documents can be considered *record entries* in a file. Lamb at pp. 9, 28. Because such fields are replicated from the hub central *server* to client *servers*, such as laptops or PDAs, these changes create a new second *record entry* as recited in claim 2. Lamb at pp. 9, 28. Such new *record entry* on the client *server* is replicated from the central hub *server* and so is based on the new third *record entry* as recited in claim 2. Lamb at pp. 9, 28.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

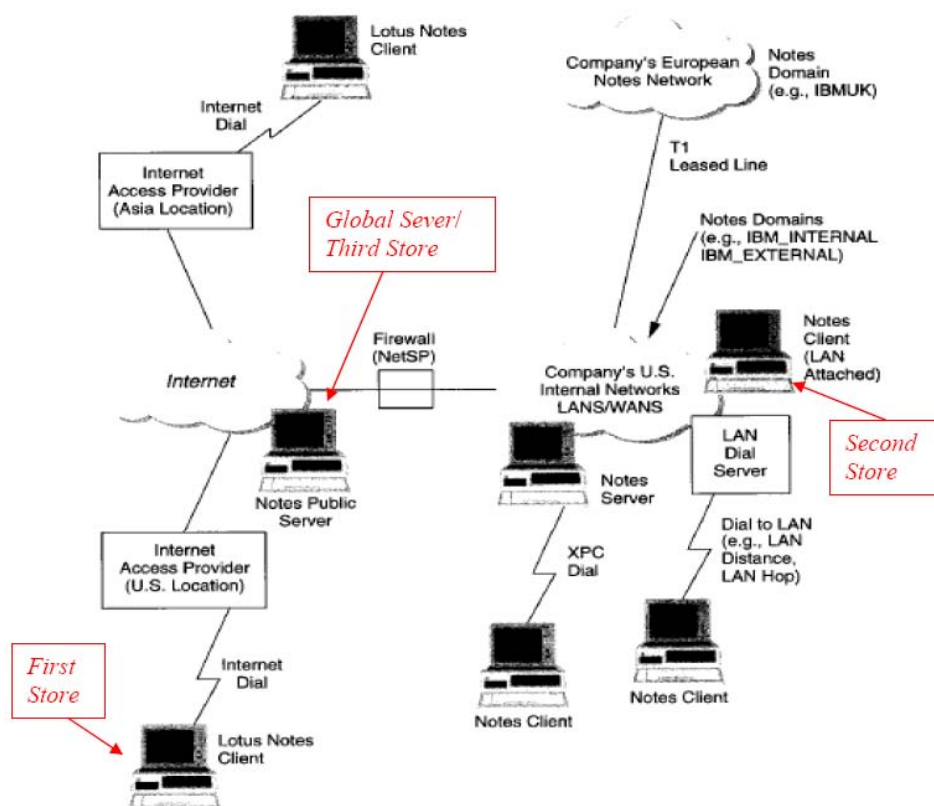
means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Lamb.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Lamb discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lamb at pp. 7, 178 at Fig. 2.1. Specifically, Lamb discloses a “hub and spoke” concept having a Lotus Notes public *server* connected in a network with various Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Therefore, a *first memory store* can be one of the Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Lamb also discloses that a client can include a folder containing a Lotus Notes database with modifiable fields. Lamb at 28. Thus, the client computers of Lamb include a first folder for storing first *record entries*.



Lamb Fig. 2.1

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lamb also discloses providing a second

memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. As discussed above, Lamb discloses a network with a hub and spoke architecture, with the Lotus Notes *server* connected with client computers, and replicating to these client computers. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. The client computers can be mobile computers such as laptops or notebooks, as well as PDAs. Lamb at pp. 9, 27-28, 178, 196. One laptop or PDA can be a *first memory store* in the network, and a different laptop or PDA can be a *second memory store*. Lamb at pp. 9, 27-28, 178, 196. Because they are connected in a network, they are therefore “coupled” as recited in claim 4. Lamb at pp. 9, 27-28, 178, 196 and Fig 2.1.

These laptops or PDAs can include a folder containing a database for Lotus Notes as well as Notes applications. Lamb at pp. 9, 27-28, 178, 196. Thus, a second laptop or PDA in the network includes a folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196. Furthermore, Lamb discloses that Lotus Notes documents can be saved in the database on the *server* and client computers in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Thus, the documents are “files” according to claim 3, and the updatable fields are *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

Lamb also discloses the second folder being *synchronized* with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Lamb at pp. 9, 27-28, 178, 196. In the hub and spoke network disclosed by Lamb, the hub, also known as the Lotus Notes *server*, replicates to the spokes, also known as the Lotus Notes client computers. Lamb also discloses replicating only the modified fields of a larger database document in order to reduce the amount of data transferred. Lamb at pp. 9, 27-28, 178, 196. As stated in Lamb, with “the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*. Lamb at pp. 9, 27-28, 178, 196. Then the hub *server* will replicate to all other spoke *servers*.” Lamb at pp. 9, 27-28, 178, 196. Therefore, a first client computer can replicate to a central hub *server*, which then can replicate that data to a second client computer. Lamb at pp. 9, 27-28, 178, 196. Thus, at a first moment in

time, Lamb discloses that the content of the documents stored in the multiple client computers will be *synchronized*. Lamb at pp. 9, 27-28, 178, 196.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Lamb discloses providing a *global server* including a third memory means having a third folder for storing third *record entries* in a file. Lamb at pp. 27, 178. As discussed above, Lamb discloses a hub and spoke network architecture. Lamb at pp. 27, 178. The hub acts as a central Lotus Notes *server* and thus can be considered a *global server* including a third memory means as recited in claim 3. Lamb at pp. 27, 178.

Additionally, the central hub *server* can have saved thereto “master copies of each database.” Lamb at pp. 27, 178. Furthermore, as noted above, Lotus Notes documents can be saved in the database on the central *server* in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Therefore, the central hub has a third folder for storing third *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Lamb discloses a means for receiving new *record entries* in a file at the *first memory store*. Lamb at pp. 5-6, 28. In particular, Lamb discloses that a User can do work on a Lotus Notes client, independent of the Notes *server*. Lamb at pp. 5-6, 28. A *server*, such as a laptop or PDA on which Users work, according to Lamb, can be considered a “means” as recited in claim 3. Additionally, Release 4 of Notes enables “updated fields” across the network. Lamb at pp. 5-6, 28. Thus, changes can be made to “fields” in the Lotus Notes documents. Lamb at pp. 5-6, 28. Changes to such “fields” are receiving new first *record entries* in a file according to claim 3. Lamb at pp. 5-6, 28.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Lamb discloses a means for electronically *synchronizing* the first folder and a third folder. Lamb at pp. 9, 27-28, 178. Specifically, Lamb discloses that “[w]ith the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*...[t]hen the hub *server* will replicate to all

other *servers*.” Lamb at pp. 9, 27-28, 178. Therefore, Lamb discloses that changes made to documents on the client spoke *servers* are replicated to the central hub *servers*. Lamb at pp. 9, 27-28, 178. Because the client *server* on which changes are made contains the “first folder,” and the central hub *server* contains the “third folder,” therefore in Lamb, a first folder on a client *server* is *synchronized* with the third folder on the central hub *server* as recited in claim 3. Lamb at pp. 9, 27-28, 178.

Lamb discloses that synchronization is such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lamb at pp. 9, 27-28, 178. Release 4 of Lotus Notes enables updated “fields” in documents across the network. Lamb at pp. 9, 27-28, 178. Such fields are *record entries*. Lamb at pp. 9, 27-28, 178. Therefore, when the fields are replicated to the central hub, this discloses that a new third *record entry* is created and based on the first *record entry* from the client *server* as recited in claim 3. Lamb at pp. 9, 27-28, 178.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lamb discloses a means for electronically *synchronizing* the third folder and the second folder. Lamb at pp. 9, 28. In particular, Lamb discloses that after a client spoke *server* replicates to the central hub *server*, “[t]hen the hub *server* will replicate to all other spoke *servers*.” Therefore, after replicating to the central *server*, the changes are then replicated to other client computers. Thus, electronically *synchronizing* the third and the second folder as recited in claim 3 is disclosed.

Lamb also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. Lamb at pp. 9, 28. Furthermore, Lamb discloses that Release 4 of Lotus Notes enables updated fields across the network. Lamb at pp. 9, 28. As discussed above, such fields in the Lotus Notes documents can be considered *record entries* in a file. Lamb at pp. 9, 28. Because such fields are replicated from the hub central *server* to client *servers*, such as laptops or PDAs, these changes create a new second *record entry* as recited in claim 3. Lamb at pp. 9, 28. Such new *record entry* on

the client *server* is replicated from the central hub *server* and so is based on the new third *record entry* as recited in claim 3. Lamb at pp. 9, 28.

4. Claim 4

Claim 4 of the '231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

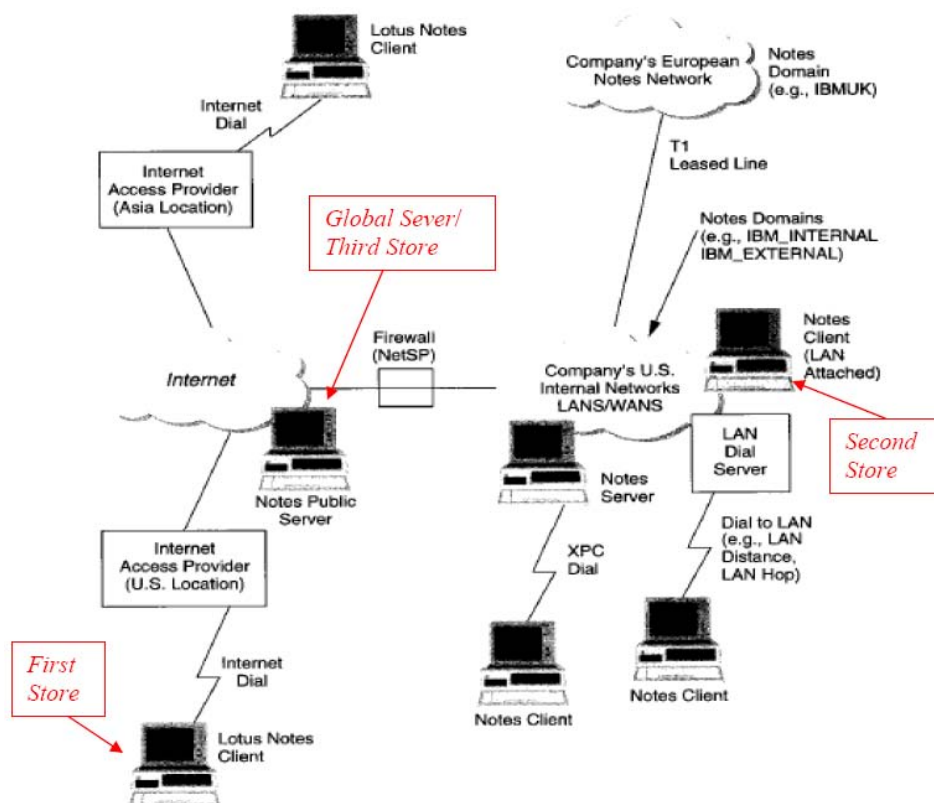
receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Lamb.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Lamb discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lamb at pp. 7, 178 at Fig. 2.1. Specifically, Lamb discloses a “hub and spoke” concept having a Lotus Notes public *server* connected in a network with various Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Therefore, a *first memory store* can be one of the Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Lamb also discloses that a client can include a folder containing a Lotus Notes database with modifiable fields. Lamb at 28. Thus, the client computers of Lamb include a first folder for storing first *record entries*.



Lamb Fig. 2.1

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lamb also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. As discussed above, Lamb discloses a network with a hub and spoke architecture, with the Lotus Notes *server* connected with client computers, and replicating to these client computers. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. The client computers can be mobile computers such as laptops or notebooks, as well as PDAs. Lamb at pp. 9, 27-28, 178, 196. One laptop or PDA can be a *first memory store* in the network, and a different laptop or PDA can be a *second memory store*. Lamb at pp. 9, 27-28, 178, 196. Because they are connected in a network, they are therefore “coupled” as recited in claim 4. Lamb at pp. 9, 27-28, 178, 196 and Fig 2.1.

These laptops or PDAs can include a folder containing a database for Lotus Notes as well as Notes applications. Lamb at pp. 9, 27-28, 178, 196. Thus, a second laptop or PDA in the network includes a folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196. Furthermore, Lamb discloses that Lotus Notes documents can be saved in the database on the *server* and client computers in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Thus, the documents are “files” according to claim 4, and the updatable fields are *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

Lamb also discloses the second folder being *synchronized* with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Lamb at pp. 9, 27-28, 178, 196. In the hub and spoke network disclosed by Lamb, the hub, also known as the Lotus Notes *server*, replicates to the spokes, also known as the Lotus Notes client computers. Lamb also discloses replicating only the modified fields of a larger database document in order to reduce the amount of data transferred. Lamb at pp. 9, 27-28, 178, 196. As stated in Lamb, with “the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*. Lamb at pp. 9, 27-28, 178, 196. Then the hub *server* will replicate to all other spoke *servers*.” Lamb at pp. 9, 27-28, 178, 196. Therefore, a first client computer can replicate to a central hub *server*, which then can replicate that data to a second client computer. Lamb at pp. 9, 27-28, 178, 196. Thus, at a first moment in time, Lamb discloses that the content of the documents stored in the multiple client computers will be *synchronized*. Lamb at pp. 9, 27-28, 178, 196.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lamb discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lamb at pp. 27, 178. As discussed above, Lamb discloses a hub and spoke network architecture. Lamb at pp. 27, 178. The hub acts as a central Lotus Notes *server* and thus can be considered a *global server* including a *third memory store* as recited in claim 4. Lamb at pp. 27, 178.

Additionally, the central hub *server* can have saved thereto “master copies of each database.” Lamb at pp. 27, 178. Furthermore, as noted above, Lotus Notes documents can be saved in the database on the central *server* in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Therefore, the central hub has a third folder for storing third *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

The fourth element of claim 4 is “receiving changes at the first memory store.” Lamb discloses receiving changes at the *first memory store*. Lamb at pp. 5-6, 28. In particular, Lamb discloses that a User can do work on a Lotus Notes client, such as a laptop or PDA, independent of the Notes *server*. Lamb at pp. 5-6, 28. Because the laptop or PDA has a *first memory store*, and changes are made thereto, Lamb discloses the abovementioned limitation.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lamb discloses electronically *synchronizing* the first folder and the third folder. Lamb at pp. 9, 28. Specifically, Lamb discloses that “[w]ith the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*...[t]hen the hub *server* will replicate to all other *servers*.” Lamb at pp. 9, 28. Therefore, Lamb discloses that changes made to documents on the client spoke *servers* are replicated to the central hub *servers*. Lamb at pp. 9, 28. Because the client *server* on which changes are made contains the “first folder,” and the central hub *server* contains the “third folder,” therefore in Lamb, a first folder on a client *server* is *synchronized* with the third folder on the central hub *server* as recited in claim 4. Lamb at pp. 9, 28.

Lamb discloses that synchronization is such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lamb at pp. 9, 28. Release 4 of Lotus Notes enables updated “fields” in documents across the network. Such fields are *record entries*. Therefore, when the fields are replicated to the central hub, this discloses that a new third *record entry* is created and based on the first *record entry* from the client *server* as recited in claim 4.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lamb discloses electronically *synchronizing* the third folder and the second folder. Lamb at pp. 9, 28. In particular, Lamb discloses that after a client spoke *server* replicates to the central hub *server*, “[t]hen the hub *server* will replicate to all other spoke *servers*.” Therefore, after replicating to the central *server*, the changes are then replicated to other client computers. Thus, electronically *synchronizing* the third and the second folder as recited in claim 4 is disclosed.

Lamb also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. Furthermore Lamb discloses that Release 4 of Lotus Notes enables updated fields across the network. As discussed above, such fields in the Lotus Notes documents can be considered “*record entries* in a file.” Because such fields are replicated from the hub central *server* to client *servers*, such as laptops or PDAs, these changes create a “new second entry” as recited in claim 4. Such new *record entry* on the client *server* is replicated from the central hub *server* and so is based on the new third *record entry* also as recited in claim 4.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

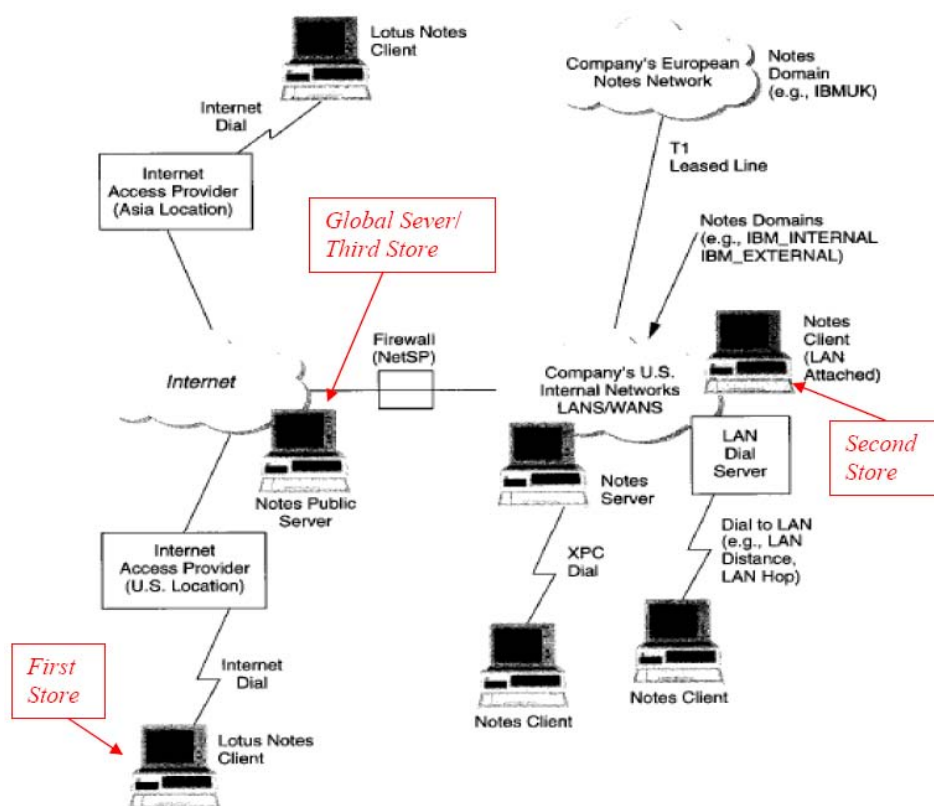
an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Lamb.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Lamb discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lamb at pp. 7, 178 at Fig. 2.1. Specifically, Lamb discloses a “hub and spoke” concept having a Lotus Notes public *server* connected in a network with various Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Therefore, a *first memory store* can be one of the Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Lamb also discloses that a client can include a folder containing a Lotus Notes database with modifiable fields. Lamb at 28. Thus, the client computers of Lamb include a first folder for storing first *record entries*.



Lamb Fig. 2.1

The second element of claim 5 is “a second memory store coupled to the first

memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lamb also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. As discussed above, Lamb discloses a network with a hub and spoke architecture, with the Lotus Notes *server* connected with client computers, and replicating to these client computers. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. The client computers can be mobile computers such as laptops or notebooks, as well as PDAs. Lamb at pp. 9, 27-28, 178, 196. One laptop or PDA can be a *first memory store* in the network, and a different laptop or PDA can be a *second memory store*. Lamb at pp. 9, 27-28, 178, 196. Because they are connected in a network, they are therefore “coupled” as recited in claim 4. Lamb at pp. 9, 27-28, 178, 196 and Fig 2.1.

These laptops or PDAs can include a folder containing a database for Lotus Notes as well as Notes applications. Lamb at pp. 9, 27-28, 178, 196. Thus, a second laptop or PDA in the network includes a folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196. Furthermore, Lamb discloses that Lotus Notes documents can be saved in the database on the *server* and client computers in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Thus, the documents are “files” according to claim 5, and the updatable fields are *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

Lamb also discloses the second folder being *synchronized* with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Lamb at pp. 9, 27-28, 178, 196. In the hub and spoke network disclosed by Lamb, the hub, also known as the Lotus Notes *server*, replicates to the spokes, also known as the Lotus Notes client computers. Lamb also discloses replicating only the modified fields of a larger database document in order to reduce the amount of data transferred. Lamb at pp. 9, 27-28, 178, 196. As stated in Lamb, with “the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*. Lamb at pp. 9, 27-28, 178, 196. Then the hub *server* will

replicate to all other spoke *servers*.” Lamb at pp. 9, 27-28, 178, 196. Therefore, a first client computer can replicate to a central hub *server*, which then can replicate that data to a second client computer. Lamb at pp. 9, 27-28, 178, 196. Thus, at a first moment in time, Lamb discloses that the content of the documents stored in the multiple client computers will be *synchronized*. Lamb at pp. 9, 27-28, 178, 196.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lamb discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lamb at pp. 27, 178. As discussed above, Lamb discloses a hub and spoke network architecture. Lamb at pp. 27, 178. The hub acts as a central Lotus Notes *server* and thus can be considered a *global server* including a *third memory store* as recited in claim 5. Lamb at pp. 27, 178.

Additionally, the central hub *server* can have saved thereto “master copies of each database.” Lamb at pp. 27, 178. Furthermore, as noted above, Lotus Notes documents can be saved in the database on the central *server* in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Therefore, the central hub has a third folder for storing third *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Lamb discloses an interface for receiving new changes at the *first memory store*. Lamb at pp. 5-6, 28. In particular, Lamb discloses that a User can do work on a Lotus Notes client, independent of the Notes *server*. Lamb at pp. 5-6, 28. A *server*, such as a laptop or PDA on which Users work, according to Lamb, can be considered “an interface” as recited in claim 5. Furthermore, because the laptop or PDA has a *first memory store*, and changes are made thereto, Lamb discloses the abovementioned limitation.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lamb discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Lamb at pp. 9,

27-28, 178. Specifically, Lamb discloses that “[w]ith the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*...[t]hen the hub *server* will replicate to all other *servers*.” Lamb at pp. 9, 27-28, 178. Therefore, Lamb discloses that changes made to documents on the client spoke *servers* are replicated to the central hub *servers*. Lamb at pp. 9, 27-28, 178. Because the client *server* on which changes are made contains the “first folder,” and the central hub *server* contains the “third folder,” therefore in Lamb, a first folder on a client *server* is *synchronized* with the third folder on the central hub *server* as recited in claim 5. Lamb at pp. 9, 27-28, 178. Additionally, the “general synchronization module” can be either the central *server* or *servers*, laptops, or PDAs on which the Users work. Lamb at pp. 9, 27-28, 178.

Lamb discloses that synchronization is such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lamb at pp. 9, 27-28, 178. Release 4 of Lotus Notes enables updated “fields” in documents across the network. Lamb at pp. 9, 27-28, 178. Such fields are *record entries*. Lamb at pp. 9, 27-28, 178. Therefore, when the fields are replicated to the central hub, this discloses that a new third *record entry* is created and based on the first *record entry* from the client *server* as recited in claim 5. Lamb at pp. 9, 27-28, 178.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lamb discloses an electronically *synchronizing* agent for electronically *synchronizing* the third folder and the second folder. Lamb at pp. 9, 28. In particular, Lamb discloses that after a client spoke *server* replicates to the central hub *server*, “[t]hen the hub *server* will replicate to all other spoke *servers*.” Therefore, after replicating to the central *server*, the changes are then replicated to other client computers. Thus, “electronically *synchronizing* the third and the second folder” as recited in claim 1 is disclosed. Furthermore, the “synchronization agent” can be either the central *server* or *servers* on which the Users work. Lamb at pp. 9, 28.

Lamb also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. Lamb at pp. 9, 28. Furthermore, Lamb discloses

that Release 4 of Lotus Notes enables updated fields across the network. Lamb at pp. 9, 28. As discussed above, such fields in the Lotus Notes documents can be considered *record entries* in a file. Lamb at pp. 9, 28. Because such fields are replicated from the hub central *server* to client *servers*, such as laptops or PDAs, these changes create a “new second entry” as recited in claim 5. Lamb at pp. 9, 28. Such new *record entry* on the client *server* is replicated from the central hub *server* and so is based on the new third *record entry* as recited in claim 5. Lamb at pp. 9, 28.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

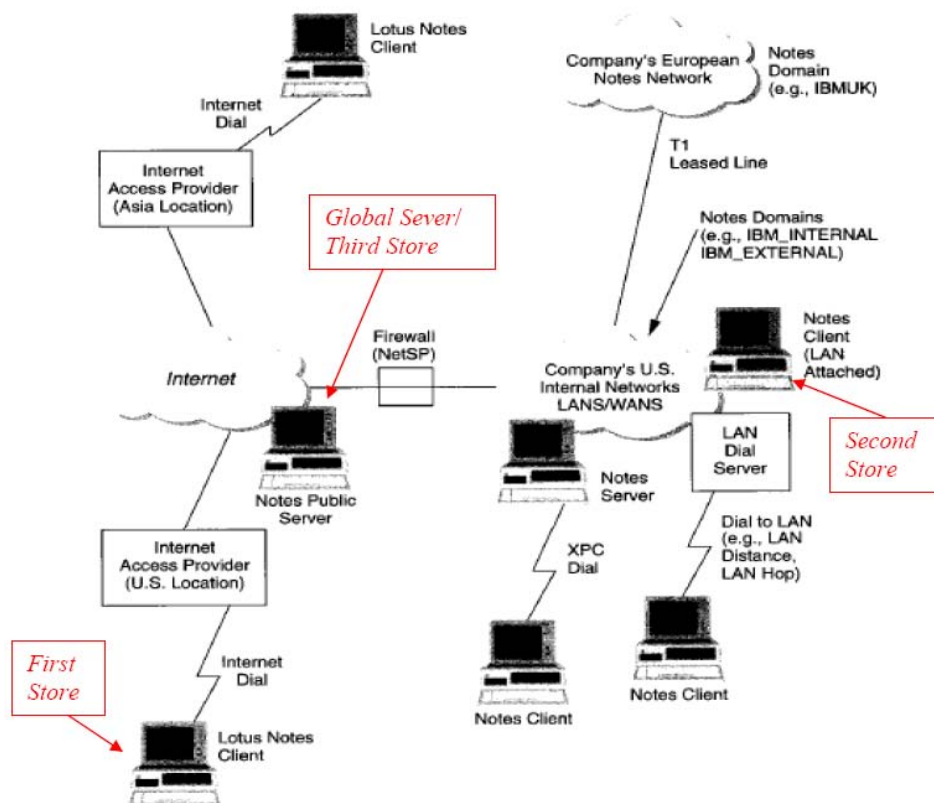
means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Lamb.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Lamb discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lamb at pp. 7, 178 at Fig. 2.1. Specifically, Lamb discloses a “hub and spoke” concept having a Lotus Notes public *server* connected in a network with various Lotus Notes client computers. Lamb at pp. 7, 178 at Fig. 2.1. Therefore, a *first memory store* can be one of the Lotus Notes client

computers. Lamb at pp. 7, 178 at Fig. 2.1. Lamb also discloses that a client can include a folder containing a Lotus Notes database with modifiable fields. Lamb at 28. Thus, the client computers of Lamb include a first folder for storing first *record entries*.



Lamb Fig. 2.1

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lamb also discloses providing a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. As discussed above, Lamb discloses a network with a hub and spoke architecture, with the Lotus Notes *server* connected with client computers, and replicating to these client computers. Lamb at pp. 9, 27-28, 178, 196 and Fig. 2.1. The client computers can be mobile computers such as laptops or notebooks, as well as PDAs.

Lamb at pp. 9, 27-28, 178, 196. One laptop or PDA can be a *first memory store* in the network, and a different laptop or PDA can be a *second memory store*. Lamb at pp. 9, 27-28, 178, 196. Because they are connected in a network, they are therefore “coupled” as recited in claim 4. Lamb at pp. 9, 27-28, 178, 196 and Fig 2.1.

These laptops or PDAs can include a folder containing a database for Lotus Notes as well as Notes applications. Lamb at pp. 9, 27-28, 178, 196. Thus, a second laptop or PDA in the network includes a folder for storing second *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196. Furthermore, Lamb discloses that Lotus Notes documents can be saved in the database on the *server* and client computers in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Thus, the documents are “files” according to claim 6, and the updatable fields are *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

Lamb also discloses the second folder being *synchronized* with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Lamb at pp. 9, 27-28, 178, 196. In the hub and spoke network disclosed by Lamb, the hub, also known as the Lotus Notes *server*, replicates to the spokes, also known as the Lotus Notes client computers. Lamb also discloses replicating only the modified fields of a larger database document in order to reduce the amount of data transferred. Lamb at pp. 9, 27-28, 178, 196. As stated in Lamb, with “the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*. Lamb at pp. 9, 27-28, 178, 196. Then the hub *server* will replicate to all other spoke *servers*.” Lamb at pp. 9, 27-28, 178, 196. Therefore, a first client computer can replicate to a central hub *server*, which then can replicate that data to a second client computer. Lamb at pp. 9, 27-28, 178, 196. Thus, at a first moment in time, Lamb discloses that the content of the documents stored in the multiple client computers will be *synchronized*. Lamb at pp. 9, 27-28, 178, 196.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Lamb discloses providing a *global server* including a third memory means having a third folder for storing third *record entries* in a file. Lamb at pp. 27, 178. As discussed above, Lamb

discloses a hub and spoke network architecture. Lamb at pp. 27, 178. The hub acts as a central Lotus Notes *server* and thus can be considered a *global server* including a third memory means as recited in claim 6. Lamb at pp. 27, 178.

Additionally, the central hub *server* can have saved thereto “master copies of each database.” Lamb at pp. 27, 178. Furthermore, as noted above, Lotus Notes documents can be saved in the database on the central *server* in the network, and additionally, fields within the documents can be updated across the network. Lamb at pp. 9, 27-28, 178, 196. Therefore, the central hub has a third folder for storing third *record entries* in a file. Lamb at pp. 9, 27-28, 178, 196.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Lamb discloses an interface for receiving new changes at the *first memory store*. Lamb at pp. 5-6, 28. In particular, Lamb discloses that a User can do work on a Lotus Notes client, independent of the Notes *server*. Lamb at pp. 5-6, 28. Furthermore, because the laptop or PDA has a *first memory store*, and changes are made thereto, Lamb discloses the abovementioned limitation. Lamb at pp. 5-6, 28.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lamb discloses a means for electronically *synchronizing* the first folder and a third folder. Lamb at pp. 9, 27-28, 178. Specifically, Lamb discloses that “[w]ith the hub-and-spoke architecture, a single spoke *server* needs only to replicate once with the hub *server*...[t]hen the hub *server* will replicate to all other *servers*.” Lamb at pp. 9, 27-28, 178. Therefore, Lamb discloses that changes made to documents on the client spoke *servers* are replicated to the central hub *servers*. Lamb at pp. 9, 27-28, 178. Because the client *server* on which changes are made contains the “first folder,” and the central hub *server* contains the “third folder,” therefore in Lamb, a first folder on a client *server* is *synchronized* with the third folder on the central hub *server* as recited in claim 6. Lamb at pp. 9, 27-28, 178.

Lamb discloses that synchronization is such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lamb at pp. 9, 27-28, 178. Release 4 of Lotus Notes enables updated “fields” in documents across the network.

Lamb at pp. 9, 27-28, 178. Such fields are *record entries*. Lamb at pp. 9, 27-28, 178. Therefore, when the fields are replicated to the central hub, this discloses that a new third *record entry* is created and based on the first *record entry* from the client *server* as recited in claim 6. Lamb at pp. 9, 27-28, 178.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lamb discloses a means for electronically *synchronizing* the third folder and the second folder. Lamb at pp. 9, 28. In particular, Lamb discloses that after a client spoke *server* replicates to the central hub *server*, “[t]hen the hub *server* will replicate to all other spoke *servers*.” Therefore, after replicating to the central *server*, the changes are then replicated to other client computers. Thus, electronically *synchronizing* the third and the second folder as recited in claim 6 is disclosed.

Lamb also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. Lamb at pp. 9, 28. Furthermore, Lamb discloses that Release 4 of Lotus Notes enables updated fields across the network. Lamb at pp. 9, 28. As discussed above, such fields in the Lotus Notes documents can be considered *record entries* in a file. Lamb at pp. 9, 28. Because such fields are replicated from the hub central *server* to client *servers*, such as laptops or PDAs, these changes create a “new second entry” as recited in claim 6. Lamb at pp. 9, 28. Such new *record entry* on the client *server* is replicated from the central hub *server* and so is based on the new third *record entry* as recited in claim 6. Lamb at pp. 9, 28.

C. Claims 1-6 are Anticipated by Lotus Deployment Guide Under 35 U.S.C. § 102

**Please see attached Exhibit CC-C
presenting claim charts for
comparison of the Lotus Deployment
Guide with claims 1-6 of the '231
patent.**

Requester respectfully submits that claims 1-6 of the '231 patent are anticipated by Lotus Deployment Guide under 35 U.S.C. § 102(b). A claim chart applying Lotus Deployment Guide to these claims is submitted herewith as Exhibit CC-C.

1. Claim 1

Claim 1 of the '231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

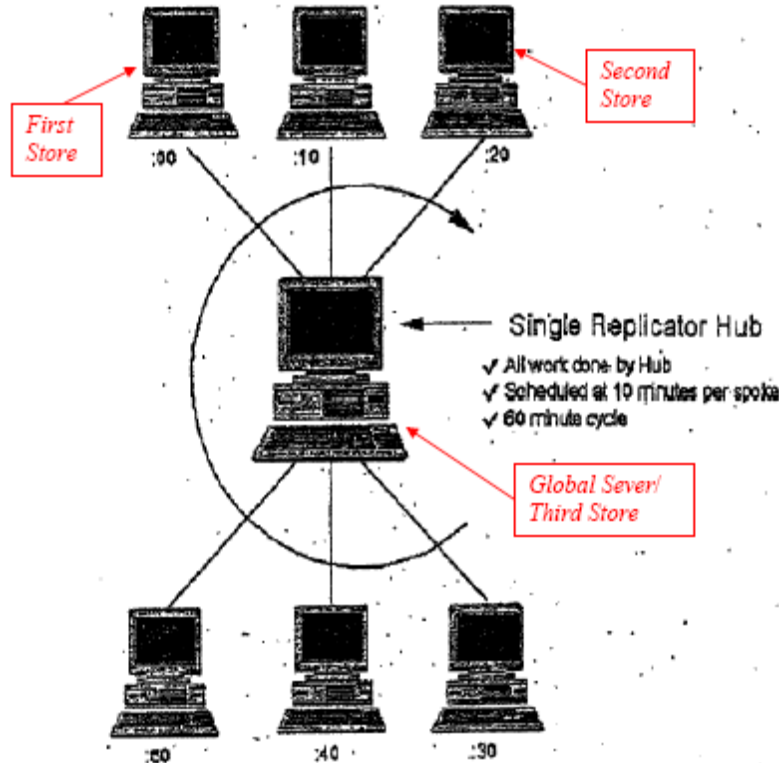
electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Lotus Deployment Guide.

The first element of claim 1 is “providing a first memory store including a first

folder for storing first record entries in a file.” Lotus Deployment Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses a *server* is able to store a database, including individual fields within documents of that database. Lotus Deployment Guide at p. 41.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Deployment Guide also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Deployment Guide at p. 41 and 51. Lotus Deployment Guide discloses a second *server* also stores a replica of the database and the two replicas are *synchronized* through replication as shown in the figure on pg. 51. Lotus Deployment Guide at p. 41 and 51.



Lotus Deployment Guide Figure on p. 51.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lotus Deployment Guide discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lotus Deployment Guide at p 48. Lotus Deployment Guide discloses one *server* is designated as a *hub server*, or *global server*, to control the replication process. Lotus Deployment Guide at p 48.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Lotus Deployment Guide discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses that *record entries* on a *server* may be added to, updated, or deleted at a later time and that these changes will be replicated to other *servers*. Lotus Deployment Guide at p. 41.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Lotus Deployment Guide discloses electronically *synchronizing* the first folder and the third folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the *hub server* replicates only the modified fields within a document with the other *servers* in sequential order, starting with the first store. Lotus Deployment Guide at pp. 41 and 51.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Deployment Guide discloses electronically *synchronizing* the third folder and the second folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the *hub server* replicates only the modified fields within a document with the other *servers* in sequential order, with replication of the second store occurring after the *hub server* replicated with the first store. Lotus Deployment Guide at pp. 41 and 51.

2. Claim 2

Claim 2 of the '231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

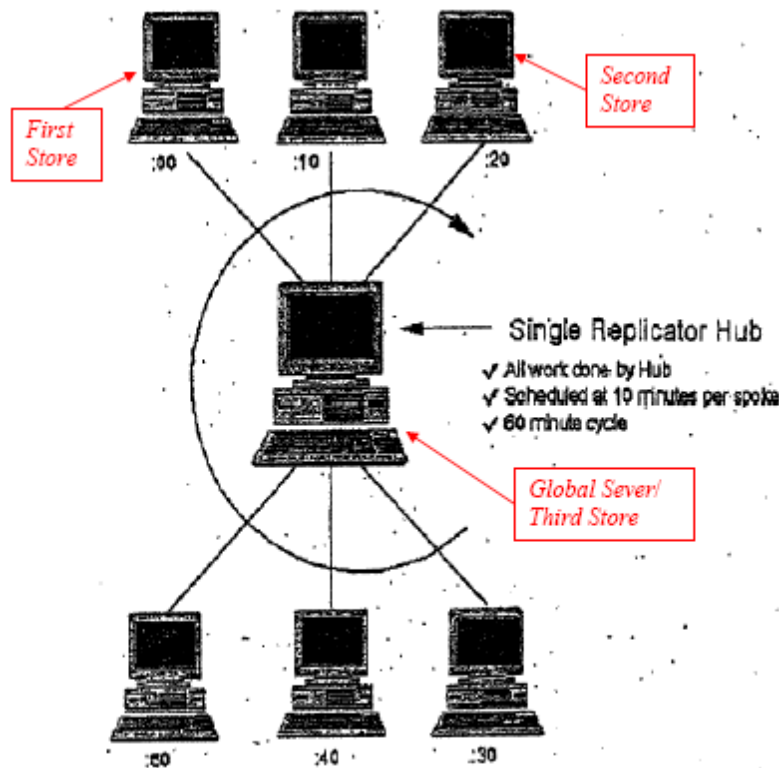
a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Lotus Deployment Guide.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Lotus Deployment Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses a *server* is able to store a database, including individual fields within documents of that database. Lotus Deployment Guide at p. 41.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Deployment Guide also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Deployment Guide at p. 41 and 51. Lotus Deployment Guide discloses a second *server* also stores a

replica of the database and the two replicas are *synchronized* through replication as shown in the figure on pg. 51. Lotus Deployment Guide at p. 41 and 51.



Lotus Deployment Guide Figure on p. 51.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Lotus Deployment Guide discloses an interface for receiving new *record entries* in a file at the *first memory store*. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses that *record entries* on a *server* may be added to, updated, or deleted at a later time and that these changes will be replicated to other *servers*. Lotus Deployment Guide at p. 41.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Lotus Deployment Guide discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, starting with the first store.

Lotus Deployment Guide at pp. 41 and 51.

Lotus Deployment Guide discloses that synchronization is such that a new third *record entry* is created in the third folder based on the new first *record entries*. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, with replication of the second store occurring after the hub *server* replicated with the first store. Lotus Deployment Guide at pp. 41 and 51.

The last element of claim 2 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Deployment Guide discloses a means for electronically *synchronizing* the third folder and the second folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, with replication of the second store occurring after the hub *server* replicated with the first store. Lotus Deployment Guide at pp. 41 and 51.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

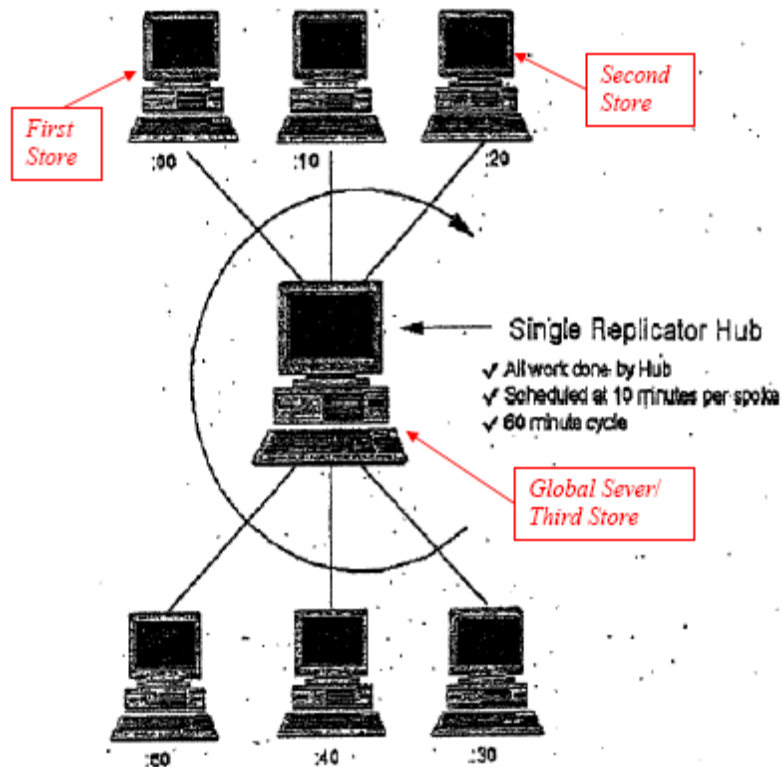
means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Lotus Deployment Guide.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Lotus Deployment Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses a *server* is able to store a database, including individual fields within documents of that database. Lotus Deployment Guide at p. 41.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Deployment Guide also discloses providing a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Deployment Guide at p. 41 and 51. Lotus Deployment Guide discloses a second *server* also stores a replica of the database and the two replicas are *synchronized* through replication as shown in the figure on pg. 51. Lotus Deployment Guide at p. 41 and 51.



Lotus Deployment Guide Figure on p. 51.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Lotus Deployment Guide discloses providing a *global server* including a third memory means having a third folder for storing third *record entries* in a file. Lotus Deployment Guide at p 48. Lotus Deployment Guide discloses one *server* is designated as a *hub server*, or *global server*, to control the replication process. Lotus Deployment Guide at p 48.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Lotus Deployment Guide discloses a means for receiving new *record entries* in a file at the *first memory store*. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses that *record entries* on a *server* may be added to, updated, or deleted at a later time and that these changes will be replicated to other *servers*. Lotus Deployment Guide at p. 41.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Lotus Deployment Guide discloses a means for

electronically *synchronizing* the first folder and a third folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, starting with the first store. Lotus Deployment Guide at pp. 41 and 51.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Deployment Guide discloses a means for electronically *synchronizing* the third folder and the second folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, with replication of the second store occurring after the hub *server* replicated with the first store. Lotus Deployment Guide at pp. 41 and 51.

4. Claim 4

Claim 4 of the ‘231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving changes at the first memory store;

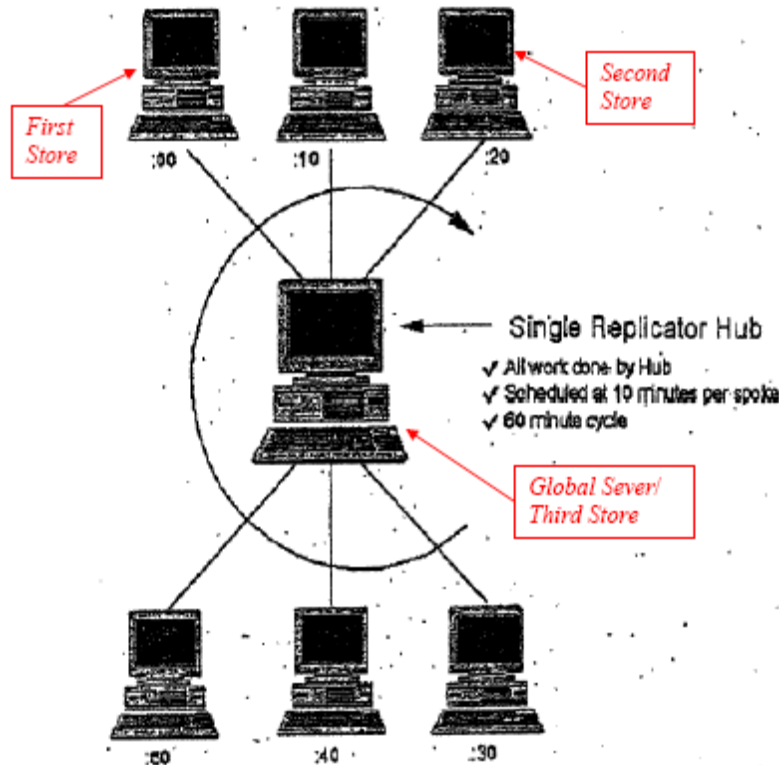
electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Lotus Deployment Guide.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Lotus Deployment Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses a *server* is able to store a database, including individual fields within documents of that database. Lotus Deployment Guide at p. 41.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Deployment Guide also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Deployment Guide at p. 41 and 51. Lotus Deployment Guide discloses a second *server* also stores a replica of the database and the two replicas are *synchronized* through replication as shown in the figure on pg. 51. Lotus Deployment Guide at p. 41 and 51.



Lotus Deployment Guide Figure on p. 51.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lotus Deployment Guide discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lotus Deployment Guide at p 48. Lotus Deployment Guide discloses one *server* is designated as a *hub server*, or *global server*, to control the replication process. Lotus Deployment Guide at p 48.

The fourth element of claim 4 is “receiving changes at the first memory store.” Lotus Deployment Guide discloses receiving changes at the *first memory store*. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses that *record entries* on a *server* may be added to, updated, or deleted at a later time and that these changes will be replicated to other *servers*. Lotus Deployment Guide at p. 41.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lotus Deployment Guide discloses electronically *synchronizing* the first folder and the third folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses

the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, starting with the first store. Lotus Deployment Guide at pp. 41 and 51.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lotus Deployment Guide discloses electronically *synchronizing* the third folder and the second folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, with replication of the second store occurring after the hub *server* replicated with the first store. Lotus Deployment Guide at pp. 41 and 51.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

an interface for receiving changes at the first memory store;

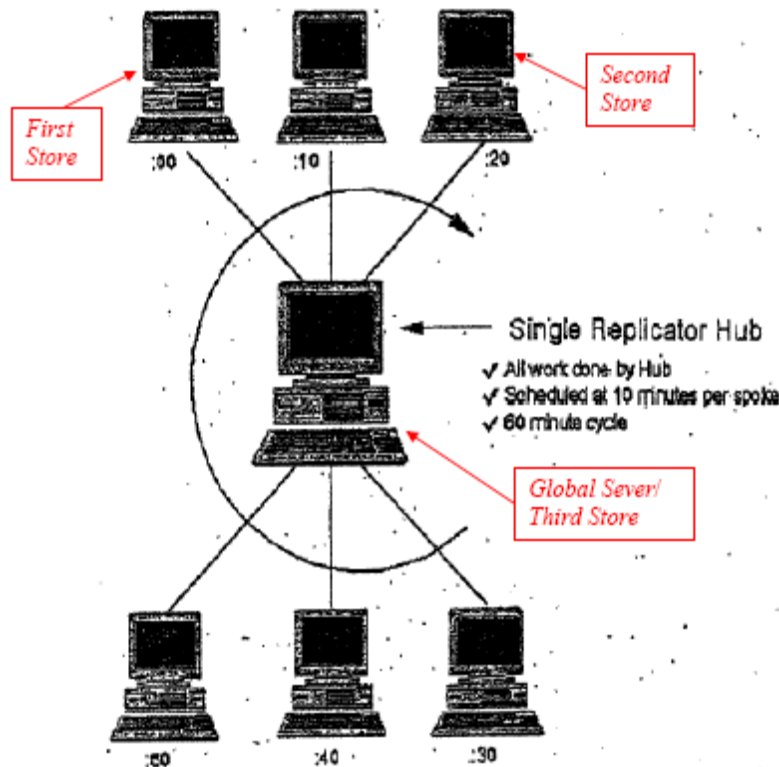
a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Lotus Deployment Guide.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Lotus Deployment Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses a *server* is able to store a database, including individual fields within documents of that database. Lotus Deployment Guide at p. 41.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Deployment Guide also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Deployment Guide at p. 41 and 51. Lotus Deployment Guide discloses a second *server* also stores a replica of the database and the two replicas are *synchronized* through replication as shown in the figure on pg. 51. Lotus Deployment Guide at p. 41 and 51.



Lotus Deployment Guide Figure on p. 51.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lotus Deployment Guide discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lotus Deployment Guide at p 48. Lotus Deployment Guide discloses one *server* is designated as a *hub server*, or *global server*, to control the replication process. Lotus Deployment Guide at p 48.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Lotus Deployment Guide discloses an interface for receiving new changes at the *first memory store*. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses that *record entries* on a *server* may be added to, updated, or deleted at a later time and that these changes will be replicated to other *servers*. Lotus Deployment Guide at p. 41.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lotus Deployment Guide discloses a general

synchronization module for electronically *synchronizing* the first folder and a third folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, starting with the first store. Lotus Deployment Guide at pp. 41 and 51.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lotus Deployment Guide discloses an electronically *synchronizing* agent for electronically *synchronizing* the third folder and the second folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, with replication of the second store occurring after the hub *server* replicated with the first store. Lotus Deployment Guide at pp. 41 and 51.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

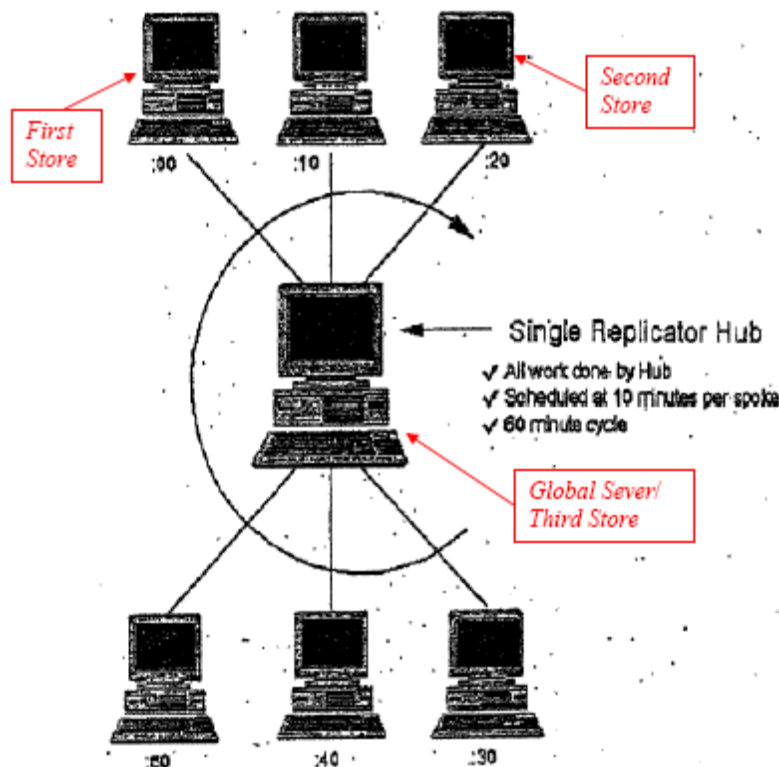
means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Lotus Deployment Guide.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Lotus Deployment Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses a *server* is able to store a database, including individual fields within documents of that database. Lotus Deployment Guide at p. 41.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Deployment Guide also discloses providing a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Deployment Guide at p. 41 and 51. Lotus Deployment Guide discloses a second *server* also stores a replica of the database and the two replicas are *synchronized* through replication as shown in the figure on pg. 51. Lotus Deployment Guide at p. 41 and 51.



Lotus Deployment Guide Figure on p. 51.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Lotus Deployment Guide discloses providing a *global server* including a third memory means having a third folder for storing third *record entries* in a file. Lotus Deployment Guide at p 48. Lotus Deployment Guide discloses one *server* is designated as a *hub server*, or *global server*, to control the replication process. Lotus Deployment Guide at p 48.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Lotus Deployment Guide discloses an interface for receiving new changes at the *first memory store*. Lotus Deployment Guide at p. 41. Lotus Deployment Guide discloses that *record entries* on a *server* may be added to, updated, or deleted at a later time and that these changes will be replicated to other *servers*. Lotus Deployment Guide at p. 41.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lotus Deployment Guide discloses a means for electronically *synchronizing* the

first folder and a third folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, starting with the first store. Lotus Deployment Guide at pp. 41 and 51.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Deployment Guide discloses a means for electronically *synchronizing* the third folder and the second folder. Lotus Deployment Guide at pp. 41 and 51. Lotus Deployment Guide discloses the hub *server* replicates only the modified fields within a document with the other *servers* in sequential order, with replication of the second store occurring after the hub *server* replicated with the first store. Lotus Deployment Guide at pp. 41 and 51.

D. Claims 1-6 are Anticipated by Lotus Admin Guide Under 35 U.S.C. § 102

**Please see attached Exhibit CC-D
presenting claim charts for
comparison of the Lotus Admin Guide
with claims 1-6 of the ‘231 patent.**

Requester respectfully submits that claims 1-6 of the ‘231 patent are anticipated by Lotus Admin Guide under 35 U.S.C. § 102(b). A claim chart applying Lotus Admin Guide to these claims is submitted herewith as Exhibit CC-D.

1. Claim 1

Claim 1 of the ‘231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

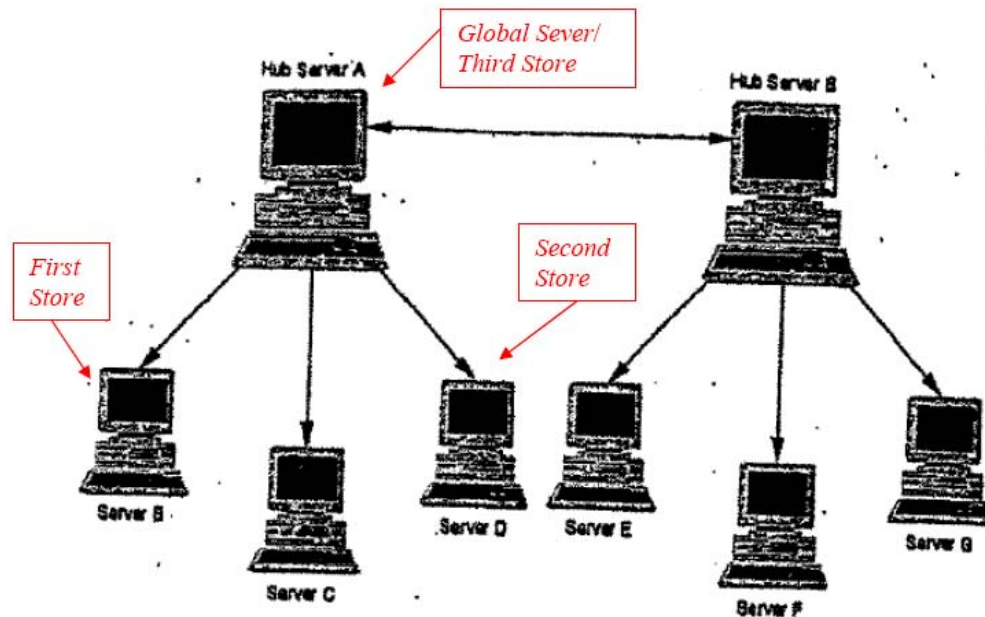
receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Lotus Admin Guide.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Lotus Admin Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Admin Guide at p. 71 and 78. Specifically, Lotus Admin Guide discloses a *server* that stores a database that includes modifiable fields within a document. Lotus Admin Guide at 71. The figure on page 78 shows how the first store is connected to the rest of the system. Lotus Admin Guide at 78.



Lotus Admin Guide Figure on p. 78.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Admin Guide also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Admin Guide at p. 71. Lotus Admin Guide discloses there are multiple *servers* with replicas so that a second *server* can be the second store. Lotus Admin Guide at 71. In addition, the first and second store are *synchronized* through a process known as replication which replicates only the modified fields instead of entire documents. Lotus Admin Guide at 71.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lotus Admin Guide discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses a hub *server* as the *global server*. Lotus Admin Guide at p. 77.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Lotus Admin Guide discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Lotus Admin Guide at p. 72. Lotus Admin Guide discloses users are able to later add, edit, or delete documents in a database on the first store. Lotus Admin Guide at p. 72.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Lotus Admin Guide discloses electronically *synchronizing* the first folder and the third folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially starting with the first store. Lotus Admin Guide at p. 77.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Admin Guide discloses electronically *synchronizing* the third folder and the second folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially, replicating with the second store after it replicates with the first store. Lotus Admin Guide at p. 77.

2. Claim 2

Claim 2 of the ‘231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

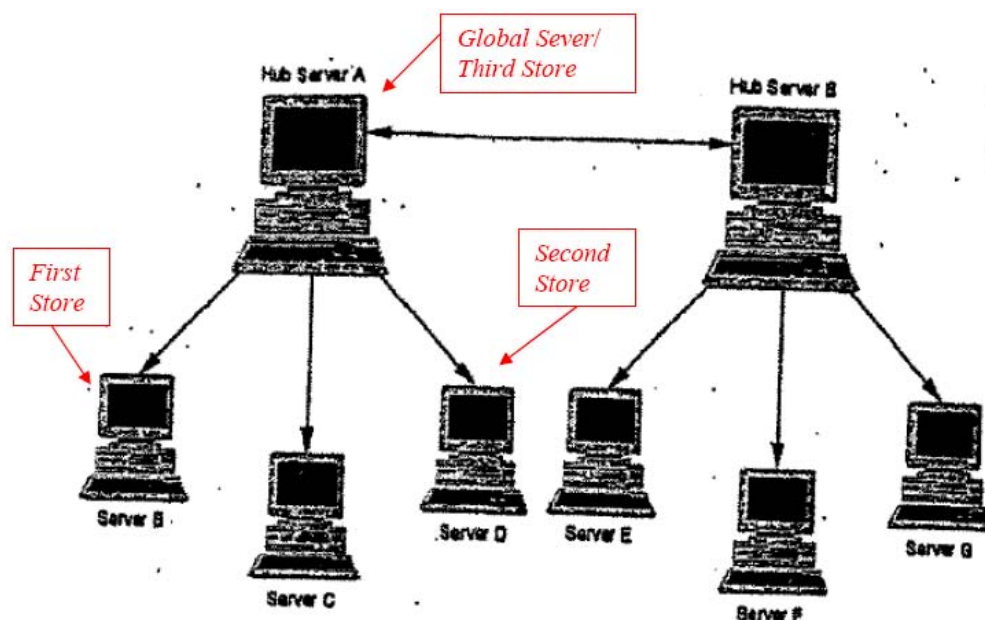
an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Lotus Admin Guide.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Lotus Admin Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Admin Guide at p. 71 and 78. Specifically, Lotus Admin Guide discloses a *server* that stores a database that includes modifiable fields within a document. Lotus Admin Guide at 71. The figure on page 78 shows how the first store is connected to the rest of the system. Lotus Admin Guide at 78.



Lotus Admin Guide Figure on p. 78.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Admin Guide also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Admin Guide at p. 71. Lotus Admin Guide discloses there are multiple *servers* with replicas so that a second *server* can be the second store. Lotus Admin Guide at 71. In addition, the first and second store are *synchronized* through a process known as replication which replicates only the modified fields instead of entire documents. Lotus Admin Guide at 71.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Lotus Admin Guide discloses an interface for receiving new *record entries* in a file at the *first memory store*. Lotus Admin Guide at p. 72. Lotus Admin Guide discloses users are able to later add, edit, or delete documents in a database on the first store. Lotus Admin Guide at p. 72.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record

entry is created in the third folder based on the new first record entries.” Lotus Admin Guide discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially starting with the first store. Lotus Admin Guide at p. 77.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Admin Guide discloses an electronically *synchronizing* agent for electronically *synchronizing* the third folder and the second folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially, replicating with the second store after it replicates with the first store. Lotus Admin Guide at p. 77.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

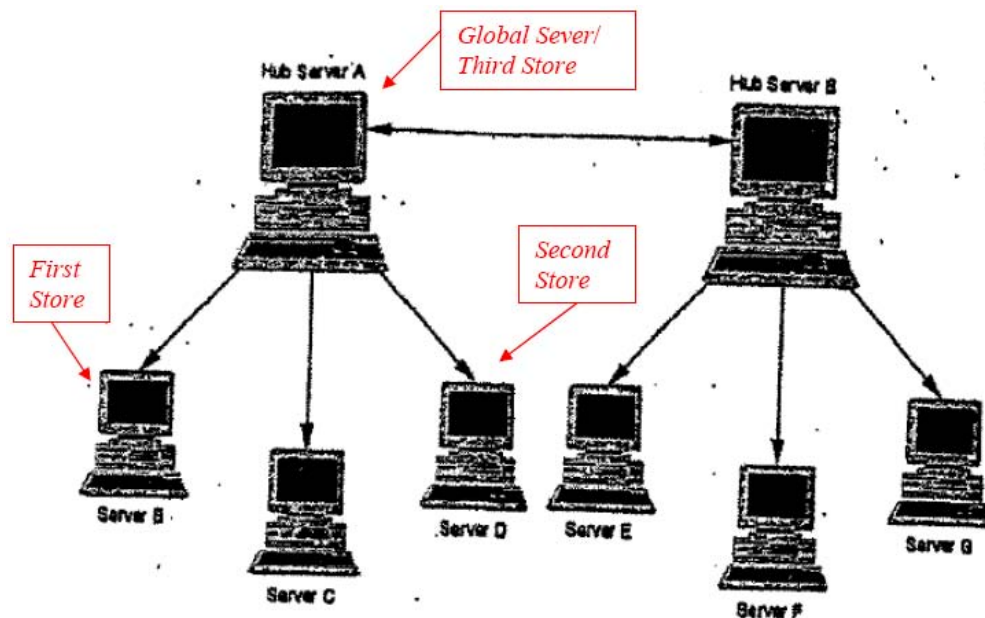
means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Lotus Admin Guide.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Lotus Admin Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Admin Guide at p. 71 and 78. Specifically, Lotus Admin Guide discloses a *server* that stores a database that includes modifiable fields within a document. Lotus Admin Guide at 71. The figure on page 78 shows how the first store is connected to the rest of the system. Lotus Admin Guide at 78.



Lotus Admin Guide Figure on p. 78.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Admin Guide also discloses providing a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Admin Guide at p. 71. Lotus Admin Guide discloses there are multiple *servers* with replicas so that a second *server* can be the second store. Lotus Admin Guide at 71. In addition, the first and second store are *synchronized* through a process known as replication which

replicates only the modified fields instead of entire documents. Lotus Admin Guide at 71.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Lotus Admin Guide discloses providing a *global server* including a third memory means having a third folder for storing third *record entries* in a file. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses a *hub server* as the *global server*. Lotus Admin Guide at p. 77.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Lotus Admin Guide discloses a means for receiving new *record entries* in a file at the *first memory store*. Lotus Admin Guide at p. 72. Lotus Admin Guide discloses users are able to later add, edit, or delete documents in a database on the first store. Lotus Admin Guide at p. 72.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Lotus Admin Guide discloses a means for electronically *synchronizing* the first folder and a third folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the *hub server* replicates with *spoke servers* sequentially starting with the first store. Lotus Admin Guide at p. 77.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Admin Guide discloses a means for electronically *synchronizing* the third folder and the second folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the *hub server* replicates with *spoke servers* sequentially, replicating with the second store after it replicates with the first store. Lotus Admin Guide at p. 77.

4. Claim 4

Claim 4 of the '231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

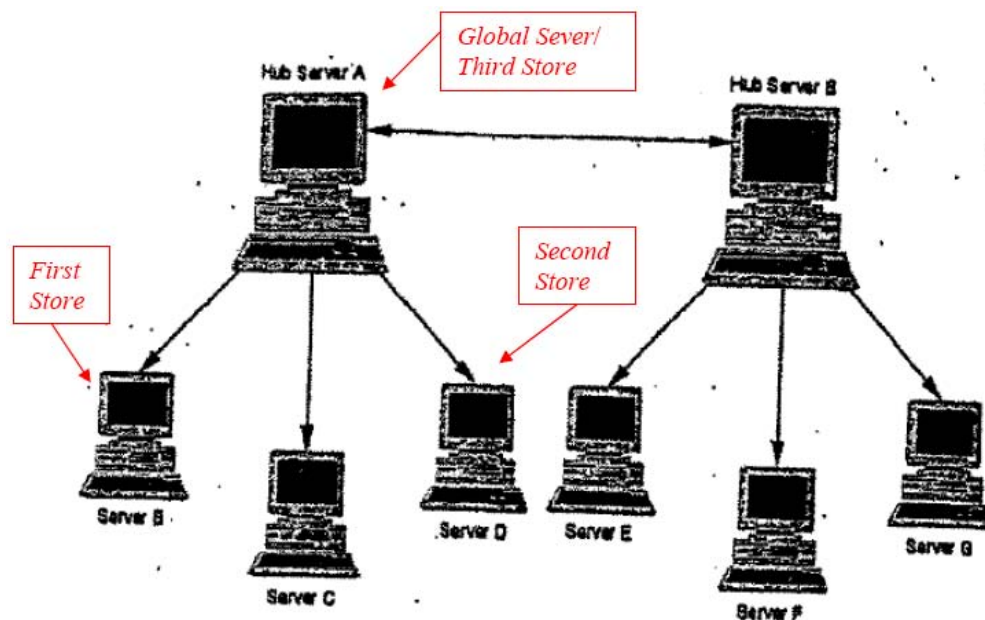
receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Lotus Admin Guide.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Lotus Admin Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Admin Guide at p. 71 and 78. Specifically, Lotus Admin Guide discloses a *server* that stores a database that includes modifiable fields within a document. Lotus Admin Guide at 71. The figure on page 78 shows how the first store is connected to the rest of the system. Lotus Admin Guide at 78.



Lotus Admin Guide Figure on p. 78.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Admin Guide also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Admin Guide at p. 71. Lotus Admin Guide discloses there are multiple *servers* with replicas so that a second *server* can be the second store. Lotus Admin Guide at 71. In addition, the first and second store are *synchronized* through a process known as replication which replicates only the modified fields instead of entire documents. Lotus Admin Guide at 71.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lotus Admin Guide discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses a *hub server* as the *global server*. Lotus Admin Guide at p. 77.

The fourth element of claim 4 is “receiving changes at the first memory store.”

Lotus Admin Guide discloses receiving changes at the *first memory store*. Lotus Admin Guide at p. 72. Lotus Admin Guide discloses users are able to later add, edit, or delete documents in a database on the first store. Lotus Admin Guide at p. 72.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lotus Admin Guide discloses electronically *synchronizing* the first folder and the third folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially starting with the first store. Lotus Admin Guide at p. 77.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lotus Admin Guide discloses electronically *synchronizing* the third folder and the second folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially, replicating with the second store after it replicates with the first store. Lotus Admin Guide at p. 77.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

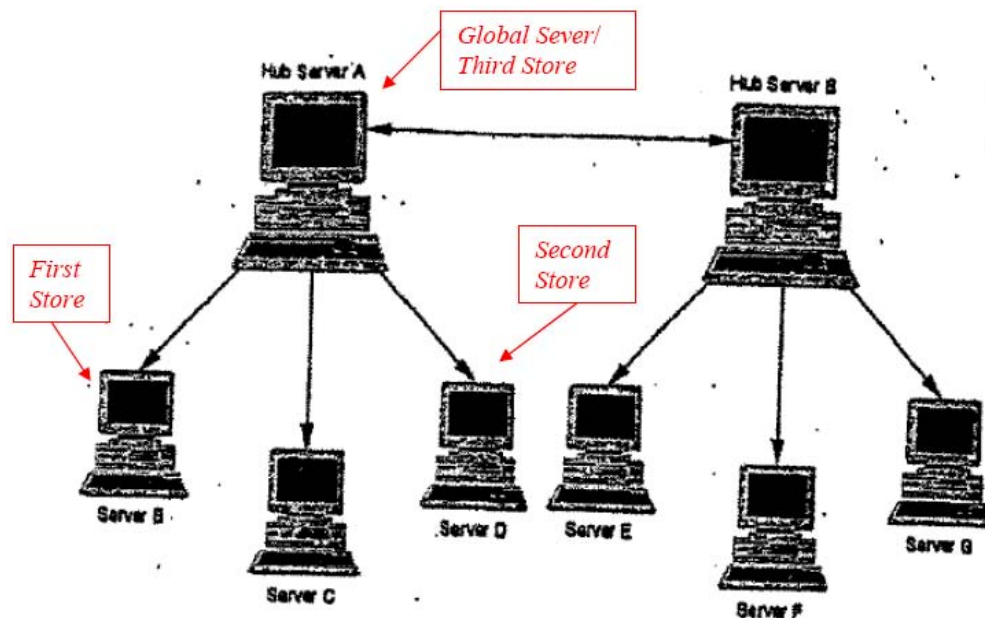
an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Lotus Admin Guide.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Lotus Admin Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Admin Guide at p. 71 and 78. Specifically, Lotus Admin Guide discloses a *server* that stores a database that includes modifiable fields within a document. Lotus Admin Guide at 71. The figure on page 78 shows how the first store is connected to the rest of the system. Lotus Admin Guide at 78.



Lotus Admin Guide Figure on p. 78.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Admin Guide also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a

second folder for storing second *record entries* in a file. Lotus Admin Guide at p. 71. Lotus Admin Guide discloses there are multiple *servers* with replicas so that a second *server* can be the second store. Lotus Admin Guide at 71. In addition, the first and second store are *synchronized* through a process known as replication which replicates only the modified fields instead of entire documents. Lotus Admin Guide at 71.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lotus Admin Guide discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses a hub *server* as the *global server*. Lotus Admin Guide at p. 77.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Lotus Admin Guide discloses an interface for receiving new changes at the *first memory store*. Lotus Admin Guide at p. 72. Lotus Admin Guide discloses users are able to later add, edit, or delete documents in a database on the first store. Lotus Admin Guide at p. 72.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lotus Admin Guide discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially starting with the first store. Lotus Admin Guide at p. 77.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lotus Admin Guide discloses an electronically *synchronizing* agent for electronically *synchronizing* the third folder and the second folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially, replicating with the second store after it replicates with the first store. Lotus Admin Guide at p. 77.

6. Claim 6

Claim 6 of the '231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

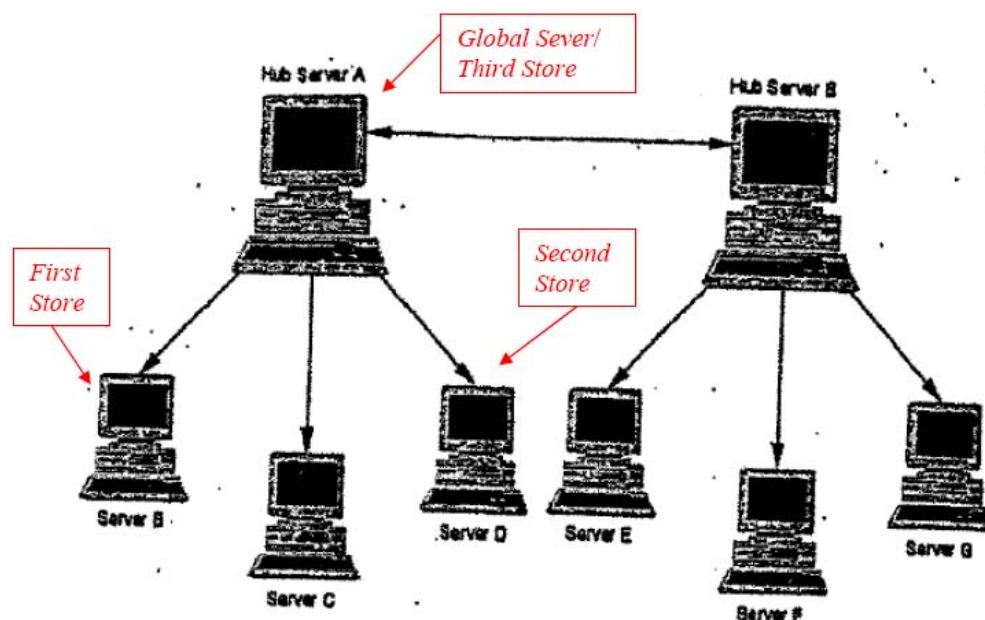
means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Lotus Admin Guide.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Lotus Admin Guide discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lotus Admin Guide at p. 71 and 78. Specifically, Lotus Admin Guide discloses a *server* that stores a database that includes modifiable fields within a document. Lotus Admin Guide at 71. The figure on page 78 shows how the first store is connected to the rest of the system. Lotus Admin Guide at 78.



Lotus Admin Guide Figure on p. 78.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lotus Admin Guide also discloses providing a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lotus Admin Guide at p. 71. Lotus Admin Guide discloses there are multiple *servers* with replicas so that a second *server* can be the second store. Lotus Admin Guide at 71. In addition, the first and second store are *synchronized* through a process known as replication which replicates only the modified fields instead of entire documents. Lotus Admin Guide at 71.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Lotus Admin Guide discloses providing a *global server* including a third memory means having a third folder for storing third *record entries* in a file. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses a *hub server* as the *global server*. Lotus Admin Guide at p. 77.

The fourth element of claim 6 is “means for receiving changes at the first memory

means.” Lotus Admin Guide discloses an interface for receiving new changes at the *first memory store*. Lotus Admin Guide at p. 72. Lotus Admin Guide discloses users are able to later add, edit, or delete documents in a database on the first store. Lotus Admin Guide at p. 72.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lotus Admin Guide discloses a means for electronically *synchronizing* the first folder and a third folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially starting with the first store. Lotus Admin Guide at p. 77.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lotus Admin Guide discloses a means for electronically *synchronizing* the third folder and the second folder. Lotus Admin Guide at p. 77. Lotus Admin Guide discloses when using the hub-and-spoke topology the hub *server* replicates with spoke *servers* sequentially, replicating with the second store after it replicates with the first store. Lotus Admin Guide at p. 77.

E. Claims 1-6 are Rendered Obvious by the Lotus Notes System Under 35 U.S.C. § 103

Please see attached Exhibit CC-A, CC-B, CC-C and CC-D presenting claim charts for comparison of the Lotus Notes System with claims 1-6 of the ‘231 patent.

Although Requester believes each Lotus Notes reference individually anticipates all claims of the ‘231 patent, in event the Examiner does not find claims 1-6 anticipated by the Lotus Notes references individually, Requester alternatively submits that claims 1-6 of the ‘231 patent are rendered obvious by Brown in view of Lamb further in view of Lotus Deployment Guide further in view of Lotus Admin Guide under 35 U.S.C. § 103. Claim charts applying Brown in view of Lamb further in view of Lotus Deployment

Guide further in view of Lotus Admin Guide to these claims is submitted herewith as Exhibits CC-M.

All four references describe with particularity the Lotus Notes system developed by IBM for *synchronizing* data among *server* and client computers. It would have been obvious to one of ordinary skill in the art to combine the teaching in all four references relating to the Lotus Notes system together in order to get a fuller understanding of the Lotus Notes system.

Claims 1-6 are rendered obvious for the same reasons as given above for the Lotus Notes references individually anticipating the claims.

F. Claims 1-6 are Anticipated by Beckhardt Under 35 U.S.C. § 102

**Please see attached Exhibit CC-E
presenting claim charts for
comparison of the Beckhardt with
claims 1-6 of the '231 patent.**

Requester respectfully submits that claims 1-6 of the '231 patent are anticipated by Beckhardt under 35 U.S.C. § 102(b). A claim chart applying Beckhardt to these claims is submitted herewith as Exhibit CC-E.

1. Claim 1

Claim 1 of the '231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Beckhardt.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Beckhardt discloses a *first memory store* as recited in claim 1. Beckhardt at Col. 1, lines 5-17. Specifically, Beckhardt discloses that multiple documents are stored separately on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Therefore, a *first memory store* can be one of the multiple computers that are connected on a network. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 1 since each document contains a plurality of fields. Beckhardt at Col. 2, lines 3-19.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Beckhardt also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file as recited in claim 1. Beckhardt at Col. 1, lines 5-17. As described above, Beckhardt discloses each document contains a plurality of fields, which are *record entries* of a file. Beckhardt at Col. 2, lines 3-19. As multiple computers that store documents are connected over a network, one computer can be considered a *first memory store*, and a second computer on the network may be considered a *second memory store*. Because they are connected over a network they are “coupled” as recited in claim 1 of Beckhardt. Beckhardt at Col. 1, lines 5-17.

Beckhardt also discloses a second folder being *synchronized* with the first folder at a first memory in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Beckhardt at Col. 1, lines 5-17. In particular, Beckhardt discloses a computing system where data is copied and stored on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Beckhardt provides Lotus notes as an example of such a system. Beckhardt at Col. 1, lines 5-17. Thus, at a first moment in time, Beckhardt discloses that the content of the documents stored in the multiple computers will be *synchronized*. Beckhardt at Col. 1, lines 5-17.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Beckhardt also discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that one copy of a document in its disclosed system, such as Lotus Notes, may be considered a “central or master copy stored on one *server* in a network.” Beckhardt at Col. 1, lines 24-32. The central *server*, which stores the central or master copy, can be considered to have a *third memory store* as recited in claim 1. Beckhardt at Col. 1, lines 24-32. As Beckhardt discloses that a document is stored on the central *server*, the reference therefore discloses that a *third memory store* has a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Other *servers* on the network store replicas of this central or master copy, and users can work with such replicas. Beckhardt at Col. 1, lines 24-32.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Beckhardt discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that replicas of the central copy are stored on other *servers* in the network, and it is these *servers* with which Users work. Beckhardt at Col. 1, lines 24-32. In particular, Beckhardt discloses that when Users modify the documents stored on the computers in the network, the changes are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Beckhardt discloses electronically *synchronizing* the first folder and the third folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. In particular, Beckhardt discloses that when Users modify the documents stored on the computers in the network, the changes are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. As stated in Beckhardt, “Users...replicate changes they make in their document replicas to the master document...” Beckhardt at Col. 1, lines 24-32. Therefore, in Beckhardt, a first folder on a User’s computer is *synchronized* with the folder on the central *server*. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses that a new third *record entry* is created in the third folder based on the new first *record entries*. As discussed above, the changes made on the User’s computer are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. Therefore, when the replication to the master document is made as disclosed in Beckhardt, a “new third *record entry* is created in the ‘third folder’” according to claim 1. Beckhardt at Col. 1, lines 24-32. Furthermore, because the changes to the master document are replicated from the User’s computers, they are therefore “based on the new first *record entries*” as recited in the present claims. Beckhardt at Col. 1, lines 24-32.

Furthermore, and in particular, Beckhardt discloses that synchronization occurs by use of sequence numbers. Beckhardt at Col. 2, lines 3-19, 32--49; Col. 3, line 63–Col. 4, line 6. For example, a document may have two fields that are revisable. Beckhardt at Col. 2, lines 3-19, 32-49. A sequence number is maintained for each document representing the number of revisions made to it, and for each document, a field sequence number for each of the revisable fields is “dynamically” maintained representing revisions to the fields. Beckhardt at Col. 2, lines 3-19, 32-49. When replication is made, the numbers can be compared to determine the most recently revised document, which can then be replicated to the corresponding field in the other documents. Beckhardt at Col. 2, lines 3-19, 32-49. Thus “synchronization” as recited in Claim 1 is disclosed by Beckhardt. Beckhardt at Col. 2, lines 3-19, 32-49.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Beckhardt discloses electronically *synchronizing* the third folder and the second folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Specifically, Beckhardt discloses that after the changes made to a User’s computer are replicated to the master document, “the master is used to pass along these changes to the other working replicas of the document.” Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Therefore, after replicating to the central *server*, the changes are then “passed” to the other computers on the network, thus “electronically *synchronizing* the third folder and the second folder” as recited in claim 1. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

Beckhardt also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. As noted above, changes replicated to the central *server* are passed to a second computer on the network. These changes passed to the second computer on the network can be considered “a new second entry” as recited in the claim. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

Additionally, because Beckhardt discloses that the changes replicated to the master document on the central *server* are “passed” to the other computers on the network, the new second entries made to a second computer on the network are based on the updated master document on the central computer. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Accordingly, this discloses the recitation in the claim that the new second *record entry* is created “based on the new third *record entry*.” Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

2. Claim 2

Claim 2 of the ‘231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder

at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Beckhardt.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Beckhardt discloses a *first memory store* as recited in claim 2. Beckhardt at Col. 1, lines 5-17. Specifically, Beckhardt discloses that multiple documents are stored separately on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Therefore, a *first memory store* can be one of the multiple computers that are connected on a network. Beckhardt at Col. 1, lines 5-17.

Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 2. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 1 since each document contains a plurality of fields. Beckhardt at Col. 2, lines 3-19.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Beckhardt also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file as recited in claim 1. Beckhardt at Col. 1, lines 5-17. As described above, Beckhardt discloses each document contains a plurality of fields, which are *record entries* of a file. Beckhardt at Col. 2, lines 3-19. As multiple computers that store documents are connected over a network, one computer can

be considered a *first memory store*, and a second computer on the network may be considered a *second memory store*. Because they are connected over a network they are “coupled” as recited in claim 1 of Beckhardt. Beckhardt at Col. 1, lines 5-17.

Beckhardt also discloses a second folder being *synchronized* with the first folder at a first memory in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Beckhardt at Col. 1, lines 5-17. In particular, Beckhardt discloses a computing system where data is copied and stored on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Beckhardt provides Lotus notes as an example of such a system. Beckhardt at Col. 1, lines 5-17. Thus, at a first moment in time, Beckhardt discloses that the content of the documents stored in the multiple computers will be *synchronized*. Beckhardt at Col. 1, lines 5-17.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Beckhardt discloses an interface for receiving new *record entries* in a file at the *first memory store*. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that replicas of the central copy are stored on other *servers* in the network, and it is these *servers* “with which Users can work.” Beckhardt at Col. 1, lines 24-32. A *server* with which Users work, according to Beckhardt, can be considered “an interface” as recited in claim 2. Beckhardt at Col. 1, lines 24-32. Beckhardt further discloses that Users are also able to “replicate changes they make in their document replicas.” Beckhardt at Col. 1, lines 24-32. Because Users can make changes to documents on the *servers* with which they work, Beckhardt therefore discloses that the “interface” can receive “new *record entries* in a file at the *first memory store*,” as recited in claim 2. Beckhardt at Col. 1, lines 24-32.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Beckhardt discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. In particular, Beckhardt discloses that when Users modify the documents stored on the

computers in the network, the changes are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. As stated in Beckhardt, “Users...replicate changes they make in their document replicas to the master document...” Beckhardt at Col. 1, lines 24-32. Therefore, in Beckhardt, a first folder on a Users computer is *synchronized* with the folder on the central *server*. Beckhardt at Col. 1, lines 24-32. Thus, the “general synchronization module” can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32. Additionally, the central *server* of Beckhardt can include a “third folder” because it can hold documents in its memory.

Beckhardt also discloses a new third *record entry* is created in the third folder based on the new first *record entries*. As discussed above, the changes made on the User’s computer are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. Therefore, when the replication to the master document is made as disclosed in Beckhardt, a “new third *record entry* is created in the ‘third folder’” according to claim 2. Beckhardt at Col. 1, lines 24-32. Furthermore, because the changes to the master document are replicated from the Users’ computers, they are therefore “based on the new first *record entries*” as recited in the present claims. Beckhardt at Col. 1, lines 24-32.

Furthermore, and in particular, Beckhardt discloses that synchronization occurs by use of sequence numbers. Beckhardt at Col. 2, lines 3-19, 32-49; Col. 3, line 63–Col. 4, line 6. For example, a document may have two fields that are revisable. Beckhardt at Col. 2, lines 3-19, 32-49. A sequence number is maintained for each document representing the number of revisions made to it, and for each document, a field sequence number for each of the revisable fields is “dynamically” maintained representing revisions to the fields. Beckhardt at Col. 2, lines 3-19, 32-49. When replication is made, the numbers can be compared to determine the most recently revised document, which can then be replicated to the corresponding field in the other documents. Beckhardt at Col. 2, lines 3-19, 32-43, 44-49. Thus “synchronization” as recited in Claim 2 is disclosed by Beckhardt. Beckhardt at Col. 2, lines 3-19, 32-49.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Beckhardt discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Specifically, Beckhardt discloses that after the changes made to a User’s computer are replicated to the master document, “the master is used to pass along these changes to the other working replicas of the document.” Beckhardt at Col. 1, lines 24-32. Therefore, after replicating to the central *server*, the changes are then “passed” to the other computers on the network, thus “electronically *synchronizing* the third folder and the second folder” as recited in claim 2. Beckhardt at Col. 1, lines 24-32. Furthermore the “synchronization agent” can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. As noted above, changes replicated to the central *server* are passed to a second computer on the network. These changes passed to the second computer on the network can be considered “a new second entry” as recited in the claim. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Additionally, because Beckhardt discloses that the changes replicated to the master document on the central *server* are “passed” to the other computers on the network, the new second entries made to a second computer on the network are based on the updated master document on the central computer. Beckhardt at Col. 1, lines 24-32. Accordingly, this discloses the recitation in the claim that the new second *record entry* is created “based on the new third *record entry*.” Beckhardt at Col. 1, lines 24-32.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder

at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Beckhardt.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Beckhardt discloses a first memory means as recited in claim 3. Beckhardt at Col. 1, lines 5-17. Specifically, Beckhardt discloses that multiple documents are stored separately on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Therefore, a first memory means can be one of the multiple computers that are connected on a network. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 3. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 1 since each document contains a plurality of fields. Beckhardt at Col. 2, lines 3-19.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Beckhardt also discloses a second memory means coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file as recited in claim 3. Beckhardt at Col. 1, lines 5-17. As described above, Beckhardt discloses each document contains a plurality of fields, which are *record entries* of a file. Beckhardt at Col. 2, lines 3-19. As

multiple computers that store documents are connected over a network, one computer can be considered a first memory means, and a second computer on the network may be considered a second memory means, and because they are connected over a network, they are “coupled” as recited in claim 3 of Beckhardt. Beckhardt at Col. 1, lines 5-17.

Beckhardt also discloses a second folder being *synchronized* with the first folder at a first moment in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Beckhardt at Col. 1, lines 5-17. In particular, Beckhardt discloses a computing system where data is copied and stored on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Beckhardt provides Lotus notes as an example of such a system. Beckhardt at Col. 1, lines 5-17. Thus, at a first moment in time, Beckhardt discloses that the content of the documents stored in the multiple computers will be *synchronized*. Beckhardt at Col. 1, lines 5-17.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Beckhardt also discloses providing a *global server* including a third memory means having a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that one copy of a document in its disclosed system, such as Lotus Notes, may be considered a “central or master copy stored on one *server* in a network.” Beckhardt at Col. 1, lines 24-32. The central *server*, which stores the central or master copy, can be considered as having a third memory means as recited in claim 3. Beckhardt at Col. 1, lines 24-32. As Beckhardt discloses that a document is stored on the central *server*, the reference therefore discloses that a *third memory store* has a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Other *servers* on the network store replicas of this central or master copy, and users can work with such replicas. Beckhardt at Col. 1, lines 24-32.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Beckhardt discloses a means for receiving new *record entries* in a file at the *first memory store*. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that replicas of the central copy are stored on other *servers* in the

network, and it is these *servers* “with which Users can work.” Beckhardt at Col. 1, lines 24-32. A *server* with which Users work, according to Beckhardt, can be considered a “means” as recited in claim 3. Beckhardt at Col. 1, lines 24-32. Beckhardt further discloses that Users are also able “replicate changes they make in their document replicas.” Beckhardt at Col. 1, lines 24-32. As Users can make changes to documents on the *servers* with which they work, Beckhardt therefore discloses the “means” can receive “new *record entries* in a file at the *first memory store*” as recited in claim 3. Beckhardt at Col. 1, lines 24-32.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Beckhardt discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. In particular, Beckhardt discloses that when Users modify the documents stored on the computers in the network, the changes are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. As stated in Beckhardt, “Users...replicate changes they make in their document replicas to the master document...” Beckhardt at Col. 1, lines 24-32. Therefore, in Beckhardt, a first folder on a User’s computer is *synchronized* with the folder on the central *server*. Beckhardt at Col. 1, lines 24-32. Thus the “general synchronization module” can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses a new third *record entry* is created in the third folder based on the new first *record entries*. As discussed above, the changes made on the User’s computer are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. Therefore, when the replication to the master document is made as disclosed in Beckhardt, a “new third *record entry* is created in the “third folder” according to claim 3. Beckhardt at Col. 1, lines 24-32. Furthermore, because the changes to the master document are replicated from the User’s computers, they are therefore “based on the new first *record entries*” as recited in the present claims. Beckhardt at Col. 1, lines 24-32.

Furthermore, and in particular, Beckhardt discloses that synchronization occurs by use of sequence numbers. Beckhardt at Col. 2, lines 3-19, 32-49; Col. 3, line 63–Col. 4, line 6. For example, a document may have two fields that are revisable. Beckhardt at Col. 2, lines 3-19, 32-49. A sequence number is maintained for each document representing the number of revisions made to it, and for each document, a field sequence number for each of the revisable fields is “dynamically” maintained representing revisions to the fields. Beckhardt at Col. 2, lines 3-19, 32-49. When replication is made, the numbers can be compared to determine the most recently revised document, which can then be replicated to the corresponding field in the other documents. Beckhardt at Col. 2, lines 3-19, 32-49. Thus “synchronization” as recited in Claim 3 is disclosed by Beckhardt. Beckhardt at Col. 2, lines 3-19, 32-49.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Beckhardt discloses a means for electronically *synchronizing* the third folder and the second folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Specifically, Beckhardt discloses that after the changes made to a User’s computer are replicated to the master document, “the master is used to pass along these changes to the other working replicas of the document.” Beckhardt at Col. 1, lines 24-32. Therefore, after replicating to the central *server*, the changes are then “passed” to the other computers on the network, thus disclosing “electronically *synchronizing* the third folder and the second folder” as recited in claim 3. Beckhardt at Col. 1, lines 24-32. Furthermore the “synchronization agent” can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. As noted above, changes replicated to the central *server* are passed to a second computer on the network. A change passed to the second computer on the network can be considered “a new second entry” as recited in the claim. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

Additionally, because Beckhardt discloses that the changes replicated to the master document on the central *server* are “passed” to the other computers on the

network, the new second entries made to a second computer on the network are based on the updated master document on the central computer. Beckhardt at Col. 1, lines 24-32. Accordingly, this discloses the recitation in the claim that the new second *record entry* is created “based on the new third *record entry*.” Beckhardt at Col. 1, lines 24-32.

4. Claim 4

Claim 4 of the ‘231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Beckhardt.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Beckhardt discloses a *first memory store* as recited in claim 4. Beckhardt at Col. 1, lines 5-17. Specifically, Beckhardt discloses that multiple documents are stored separately on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Therefore, a *first memory store* can be one of the multiple computers that are connected on a network. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 4. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder

for storing first *record entries* in a file as recited in claim 1 since each document contains a plurality of fields. Beckhardt at Col. 2, lines 3-19.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Beckhardt also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file as recited in claim 4. Beckhardt at Col. 1, lines 5-17. As described above, Beckhardt discloses each document contains a plurality of fields, which are *record entries* of a file. Beckhardt at Col. 2, lines 3-19. As multiple computers that store documents are connected over a network, one computer can be considered a *first memory store*, and a second computer on the network may be considered a *second memory store*, and because they are connected over a network they are “coupled” as recited in claim 4 of Beckhardt. Beckhardt at Col. 1, lines 5-17.

Beckhardt also discloses a second folder being *synchronized* with the first folder at a first memory in time, such that the content of the second folder is consistent with the content of the first folder at the first moment in time. Beckhardt at Col. 1, lines 5-17. In particular, Beckhardt discloses a computing system where data is copied and stored on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Beckhardt provides Lotus notes as an example of such a system. Beckhardt at Col. 1, lines 5-17. Thus, at a first moment in time, Beckhardt discloses that the content of the documents stored in the multiple computers will be *synchronized*. Beckhardt at Col. 1, lines 5-17.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Beckhardt also discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that one copy of a document in its disclosed system, such as Lotus Notes, may be considered a “central or master copy stored on one *server* in

a network.” Beckhardt at Col. 1, lines 24-32. The central *server*, which stores the central or master copy, can be considered as having a *third memory store* as recited in claim 4. Beckhardt at Col. 1, lines 24-32. Because Beckhardt discloses that a document is stored on the central *server*, the reference therefore discloses that a *third memory store* has a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Other *servers* on the network store replicas of this central or master copy, and users can work with such replicas. Beckhardt at Col. 1, lines 24-32.

The fourth element of claim 4 is “receiving changes at the first memory store.” Beckhardt discloses receiving changes at a *first memory store*. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that replicas of the central copy are stored on other *servers* in the network, and it is these *servers* with which Users work. Beckhardt at Col. 1, lines 24-32. When the Users work with such replicas on the other *servers* and make changes to the documents thereon, this discloses “receiving changes at the *first memory store*” as recited in claim 4. Beckhardt at Col. 1, lines 24-32.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Beckhardt discloses electronically *synchronizing* the first folder and the third folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. In particular, Beckhardt discloses that when Users modify the documents stored on the computers in the network, the changes are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. As stated in Beckhardt, “Users...replicate changes they make in their document replicas to the master document...” Beckhardt at Col. 1, lines 24-32. Therefore, in Beckhardt, a first folder on a User’s computer is *synchronized* with the folder on the central *server*. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses “the content of the changes creates a third *record entry*” as recited in claim 4. As discussed above, the changes made on the User’s computer are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. Therefore, when the replication to the master document is made as disclosed in Beckhardt, a “new third *record entry* is created in the “third folder” according to claim 4. Beckhardt at Col. 1, lines 24-32. Furthermore, because the changes to the master

document are replicated from the Users' computers, the content of the changes creates a third *record entry* as recited in claim 4. Beckhardt at Col. 1, lines 24-32.

Furthermore, and in particular, Beckhardt discloses that synchronization occurs by use of sequence numbers. Beckhardt at Col. 2, lines 3-19, 32-49; Col. 3, line 63–Col. 4, line 6. For example, a document may have two fields which are revisable. Beckhardt at Col. 2, lines 3-19, 32-49. A sequence number is maintained for each document representing the number of revisions made to it, and for each document, a field sequence number for each of the revisable fields is “dynamically” maintained representing revisions to the fields. Beckhardt at Col. 2, lines 3-19, 32-49. When replication is made, the numbers can be compared to determine the most recently revised document, which can then be replicated to the corresponding field in the other documents. Beckhardt at Col. 2, lines 3-19, 32-49. Thus “synchronization” as recited in Claim 4 is disclosed by Beckhardt. Beckhardt at Col. 2, lines 3-19, 32-49.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Beckhardt discloses electronically *synchronizing* the third folder and the second folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Specifically, Beckhardt discloses that after the changes made to a User's computer are replicated to the master document, “the master is used to pass along these changes to the other working replicas of the document.” Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Therefore, after replicating to the central *server*, the changes are then “passed” to the other computers on the network, thus disclosing “electronically *synchronizing* the third folder and the second folder” as recited in claim 4. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

Beckhardt also discloses a new second *record entry* is created in the second folder based on the third *record entry*. As noted above, changes replicated to the central *server* are passed to a second computer on the network. A change passed to the second computer on the network can be considered “a new second entry” as recited in the claim. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

Additionally, because Beckhardt discloses that the changes replicated to the master document on the central *server* are “passed” to the other computers on the network, the new second entries made to a second computer on the network are based on the updated master document on the central computer. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Accordingly, this discloses the recitation in the claim that the new second *record entry* is created “based on the third *record entry*.” Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Beckhardt.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Beckhardt discloses a *first memory store* as recited in claim 5. Beckhardt at Col. 1, lines 5-17. Specifically, Beckhardt discloses that multiple documents are stored separately on multiple computer systems connected over a

network. Beckhardt at Col. 1, lines 5-17. Therefore, a *first memory store* can be one of the multiple computers that are connected on a network. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 5. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 1 since each document contains a plurality of fields. Beckhardt at Col. 2, lines 3-19.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Beckhardt also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file as recited in claim 5. Beckhardt at Col. 1, lines 5-17. As described above, Beckhardt discloses each document contains a plurality of fields, which are *record entries* of a file. Beckhardt at Col. 2, lines 3-19. As multiple computers that store documents are connected over a network, one computer can be considered a *first memory store*, and a second computer on the network may be considered a *second memory store*, and because they are connected over a network, they are “coupled” as recited in claim 5 of Beckhardt. Beckhardt at Col. 1, lines 5-17.

Beckhardt also discloses a second folder being *synchronized* with the first folder at a first moment in time, such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Beckhardt at Col. 1, lines 5-17. In particular, Beckhardt discloses a computing system where data is copied and stored on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Beckhardt provides Lotus notes as an example of such a system. Beckhardt at Col. 1, lines 5-17. Thus, at a first moment in time, Beckhardt discloses that the content of the documents stored in the multiple computers will be *synchronized*. Beckhardt at Col. 1, lines 5-17.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Beckhardt

also discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that one copy of a document in its disclosed system, such as Lotus Notes, may be considered a “central or master copy stored on one *server* in a network.” Beckhardt at Col. 1, lines 24-32. The central *server*, which stores the central or master copy, can be considered as having a *third memory store* as recited in claim 5. Beckhardt at Col. 1, lines 24-32. As Beckhardt discloses that a document is stored on the central *server*, the reference therefore discloses that a *third memory store* has a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Other *servers* on the network store replicas of this central or master copy, and users can work with such replicas. Beckhardt at Col. 1, lines 24-32.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Beckhardt discloses an interface for receiving new *record entries* in a file at the *first memory store*. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that replicas of the central copy are stored on other *servers* in the network, and it is these *servers* “with which Users can work.” Beckhardt at Col. 1, lines 24-32. A *server* with which Users work, according to Beckhardt, can be considered “an interface” as recited in claim 5. Beckhardt at Col. 1, lines 24-32. Beckhardt further discloses that Users are also able “replicate changes they make in their document replicas.” Beckhardt at Col. 1, lines 24-32. Because Users can make changes to documents on the *servers* with which they work, Beckhardt therefore discloses that the “interface” can receive “new *record entries* in a file at the *first memory store*” as recited in claim 5. Beckhardt at Col. 1, lines 24-32.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Beckhardt discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. In particular, Beckhardt discloses that when Users modify the documents stored on the computers in the network, the changes are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. As stated in Beckhardt, “Users...replicate changes they make in their document replicas to

the master document...” Beckhardt at Col. 1, lines 24-32. Therefore, in Beckhardt, a first folder on a User’s computer is *synchronized* with the folder on the central *server*. Beckhardt at Col. 1, lines 24-32. Thus the “general synchronization module” can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses “the content of the changes creates a third *record entry*” as recited in claim 5. As discussed above, the changes made on the User’s computer are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. Therefore, when the replication to the master document is made as disclosed in Beckhardt, a “new third *record entry* is created in the ‘third folder’” according to claim 5. Beckhardt at Col. 1, lines 24-32. Furthermore, because the changes to the master document are replicated from the User’s computers, the content of the changes creates a third *record entry* as recited in claim 5. Beckhardt at Col. 1, lines 24-32.

Furthermore, and in particular, Beckhardt discloses that synchronization occurs by use of sequence numbers. Beckhardt at Col. 2, lines 3-19, 32-49; Col. 3, line 63–Col. 4, line 6. For example, a document may have two fields that are revisable. Beckhardt at Col. 2, lines 3-19, 32-49. A sequence number is maintained for each document representing the number of revisions made to it, and for each document, a field sequence number for each of the revisable fields is “dynamically” maintained representing revisions to the fields. Beckhardt at Col. 2, lines 3-19, 32-49. When replication is made, the numbers can be compared to determine the most recently revised document, which can then be replicated to the corresponding field in the other documents. Beckhardt at Col. 2, lines 3-19, 32-49. Thus “synchronization” as recited in Claim 5 is disclosed by Beckhardt. Beckhardt at Col. 2, lines 3-19, 32-49.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Beckhardt discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. Specifically, Beckhardt discloses that after the changes made to a User’s computer are replicated to the master document, “the master is used to pass along these changes to the other working replicas

of the document.” Beckhardt at Col. 1, lines 24-32. Therefore, after replicating to the central *server*, the changes are then “passed” to the other computers on the network, thus “electronically *synchronizing* the third folder and the second folder” as recited in claim 5. Beckhardt at Col. 1, lines 24-32. Furthermore the “synchronization agent” can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. As noted above, changes replicated to the central *server* are passed to a second computer on the network. These changes passed to the second computer on the network can be considered “a new second entry” as recited in the claim. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

Additionally, because Beckhardt discloses that the changes replicated to the master document on the central *server* are “passed” to the other computers on the network, the new second entries made to a second computer on the network are based on the updated master document on the central computer. Beckhardt at Col. 1, lines 24-32. Accordingly, this discloses the recitation in the claim that the new second *record entry* is created “based on the new third *record entry*.” Beckhardt at Col. 1, lines 24-32.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Beckhardt.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Beckhardt discloses a *first memory store* as recited in claim 6. Beckhardt at Col. 1, lines 5-17. Specifically, Beckhardt discloses that multiple documents are stored separately on multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Therefore, a *first memory store* can be one of the multiple computers that are connected on a network. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 6. Beckhardt at Col. 1, lines 5-17. Beckhardt also discloses a first folder for storing first *record entries* in a file as recited in claim 1 since each document contains a plurality of fields. Beckhardt at Col. 2, lines 3-19.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Beckhardt also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file as recited in claim 6. Beckhardt at Col. 1, lines 5-17. As described above, Beckhardt discloses each document contains a plurality of fields, which are *record entries* of a file. Beckhardt at Col. 2, lines 3-19. As multiple computers that store documents are connected over a network, one computer can be considered a *first memory store*, and a second computer on the network may be considered a *second memory store*, and because they are connected over a network they are “coupled” as recited in claim 6 of Beckhardt. Beckhardt at Col. 1, lines 5-17.

Beckhardt also discloses a second folder being *synchronized* with the first folder at a first moment in time, such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Beckhardt at Col. 1, lines 5-17. In particular, Beckhardt discloses a computing system where data is copied and stored on

multiple computer systems connected over a network. Beckhardt at Col. 1, lines 5-17. Beckhardt provides Lotus notes as an example of such a system. Beckhardt at Col. 1, lines 5-17. Thus, at a first moment in time, Beckhardt discloses that the content of the documents stored in the multiple computers will be *synchronized*. Beckhardt at Col. 1, lines 5-17.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Beckhardt also discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that one copy of a document in its disclosed system, such as Lotus Notes, may be considered a “central or master copy stored on one *server* in a network.” Beckhardt at Col. 1, lines 24-32. The central *server*, which stores the central or master copy, can be considered as having a *third memory store* as recited in claim. Beckhardt at Col. 1, lines 24-32. Because Beckhardt discloses that a document is stored on the central *server*, the reference therefore discloses that a *third memory store* has a third folder for storing third *record entries* in a file. Beckhardt at Col. 1, lines 24-32. Other *servers* on the network store replicas of this central or master copy, and users can work with such replicas. Beckhardt at Col. 1, lines 24-32.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Beckhardt discloses receiving changes at a *first memory store*. Beckhardt at Col. 1, lines 24-32. Specifically, Beckhardt discloses that replicas of the central copy are stored on other *servers* in the network, and it is these *servers* with which Users work. Beckhardt at Col. 1, lines 24-32. When the Users work with such replicas on the other *servers* and make changes to the documents thereon, this discloses “receiving changes at the *first memory store*” as recited in claim 6. Beckhardt at Col. 1, lines 24-32.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Beckhardt discloses a general synchronization module for electronically *synchronizing* the first folder and a third folder. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19. In particular, Beckhardt discloses that when Users modify the documents

stored on the computers in the network, the changes are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. As stated in Beckhardt, “Users...replicate changes they make in their document replicas to the master document...” Beckhardt at Col. 1, lines 24-32. Therefore, in Beckhardt, a first folder on a User’s computer is *synchronized* with the folder on the central *server*. Beckhardt at Col. 1, lines 24-32. Thus the “general synchronization module” can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses “the content of the changes creates a third *record entry*” as recited in claim 4. As discussed above, the changes made on the User’s computer are replicated to the master document stored on the central *server*. Beckhardt at Col. 1, lines 24-32. Therefore, when the replication to the master document is made as disclosed in Beckhardt, a “new third *record entry* is created in the ‘third folder’” according to claim 6. Beckhardt at Col. 1, lines 24-32. Furthermore, because the changes to the master document are replicated from the Users’ computers, the content of the changes creates a third *record entry* as recited in claim 4. Beckhardt at Col. 1, lines 24-32.

Furthermore, and in particular, Beckhardt discloses that synchronization occurs by use of sequence numbers. Beckhardt at Col. 2, lines 3-19, 32-49; Col. 3, line 63–Col. 4, line 6. For example, a document may have two fields that are revisable. Beckhardt at Col. 2, lines 3-19, 32-49. A sequence number is maintained for each document representing the number of revisions made to it, and for each document, a field sequence number for each of the revisable fields is “dynamically” maintained representing revisions to the fields. Beckhardt at Col. 2, lines 3-19, 32-49. When replication is made, the numbers can be compared to determine the most recently revised document, which can then be replicated to the corresponding field in the other documents. Beckhardt at Col. 2, lines 3-19, 32-49. Thus “synchronization” as recited in Claim 6 is disclosed by Beckhardt. Beckhardt at Col. 2, lines 3-19, 32-43, and 44-49

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Beckhardt discloses a means for electronically *synchronizing* the third folder and the second folder. Beckhardt at Col. 1,

lines 24-32; Col. 2, lines 3-19. Specifically, Beckhardt discloses that after the changes made to a User's computer are replicated to the master document, "the master is used to pass along these changes to the other working replicas of the document." Beckhardt at Col. 1, lines 24-32. Therefore, after replicating to the central *server*, the changes are then "passed" to the other computers on the network, thus disclosing "electronically *synchronizing* the third folder and the second folder" as recited in claim 6. Beckhardt at Col. 1, lines 24-32. Furthermore the "synchronization agent" can be the central *server* or *servers* on which the Users work. Beckhardt at Col. 1, lines 24-32.

Beckhardt also discloses that a new second *record entry* is created in the second folder based on the new third *record entry*. As noted above, changes replicated to the central *server* are passed to a second computer on the network. These changes passed to the second computer on the network can be considered "a new second entry" as recited in the claim. Beckhardt at Col. 1, lines 24-32; Col. 2, lines 3-19.

Additionally, because Beckhardt discloses that the changes replicated to the master document on the central *server* are "passed" to the other computers on the network, the new second entries made to a second computer on the network are based on the updated master document on the central computer. Beckhardt at Col. 1, lines 24-32. Accordingly, this discloses the recitation in the claim that the new second *record entry* is created "based on the new third *record entry*." Beckhardt at Col. 1, lines 24-32.

G. Claims 1-6 are Anticipated by Shaheen Under 35 U.S.C. § 102

**Please see attached Exhibit CC-F
presenting claim charts for
comparison of the Shaheen with
claims 1-6 of the '231 patent.**

Requester respectfully submits that claims 1-6 of the '231 patent are anticipated by Shaheen under 35 U.S.C. § 102(b). A claim chart applying Shaheen to these claims is submitted herewith as Exhibit CC-F.

1. Claim 1

Claim 1 of the '231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

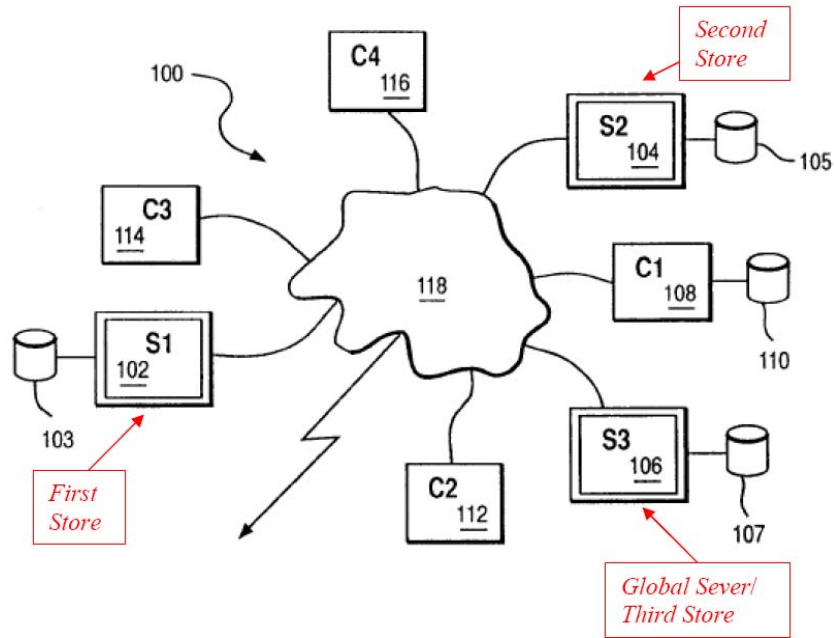
receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Shaheen.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Shaheen discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1. Specifically, Shaheen discloses a clustered or distributed system having multiple *servers*, each having non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1.



Shaheen Figure 1.

The non-volatile storage may be magnetic hard disks, optical disks or similar technology. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, Shaheen indicates that a group of these *servers* can each contain a copy of a particular fileset, where a fileset is a subtree of a file system or an information directory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. The group of *servers* having a replica of a particular fileset are known as a fileset storage group (FSG). Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Therefore, one of the multiple *servers* with the non-volatile storage in a FSG can be considered a “*first memory store*” as recited in claim 1. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, as this *server* contains a replica of a fileset, this discloses “a first folder for storing first *record entries* in a file” according to claim 1. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Shaheen at Col. 5,

lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses a clustered or distributed system having multiple *servers* with non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. When a group of these *servers* contain a replica of a particular fileset, they are part of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Therefore, one *server* in the FSG can be considered a first memory, and another *server* in the FSG can be considered a second memory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As these *servers* can store filesets, a “second folder” as recited in claim 1 is disclosed. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Additionally, the fileset replica stored on the *server* in the FSG can be considered a second *record entry* in a file according to claim 1. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

Additionally, to aid in replication, each replica will have a fileset version vector. After a modification the vector is updated, and is the new fileset version vector. This modification log is then propagated through the system during replication. Such a modification log can also be considered a *record entry* in a file.

Shaheen also discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses that an FSG will contain a particular replica of a fileset. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Furthermore, if any changes take place on the *servers*, a replication process can take place wherein all the *servers* in the FSG will again contain the same replica of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Thus before and after such replication, a first folder on one *server* in the FSG will be *synchronized* with another *server* on the FSG such that their content will be the same. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Accordingly, this discloses that at a first moment in time, the contents of “the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Shaheen also discloses a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. As noted above, Shaheen discloses multiple *servers* in an FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. When a replication or update event is to occur, all the computers in the FSG are assigned a unique priority. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. A *server* in the FSG is assigned to act as a “coordinator.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. The assignment is made such that each *server* in a FSG will have only one coordinator above it. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. This coordinator computer can be considered a *global server*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. Furthermore, as this *server* contains a non-volatile memory and a replica of a fileset, this discloses a *third memory store*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Shaheen discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Shaheen at Col. 7, lines 22-26. As disclosed in Shaheen, before replication, there is first an identification of the *servers* with replicas that are missing updates. Shaheen at Col. 7, line 22-26. Thus as some *servers* are missing updates, this indicates changes have been made to the fileset of at least one of the *servers* in the FSG. Shaheen at Col. 7, line 22-26. Thus the *server* on where the changes occurred in Shaheen can be considered as “receiving new first *record entries* in a file at the *first memory store*” according to claim 1. Shaheen at Col. 7, line 22-26. Additionally, as the changes were made to replica file sets, this therefore discloses the changes were made “at a second moment in time later than the first moment in time.” Shaheen at Col. 7, line 22-26. Additionally a modification log which notes the fileset version on the *server* is changed as well when updates are made to the replica. Such changes to the modification log can also be considered “receiving new first *record entries* in a file at the *first memory store*.” Additionally, as the modification log shows

updates made to the replica they also are made “at a second moment in time later than the first moment in time.”

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Shaheen discloses electronically *synchronizing* the first folder and the third folder. Shaheen at Col. 10, line 1-18. As noted above, when replication is to occur, each of the *servers* in the FSG is given a particular priority, and a coordinator chosen. Shaheen at Col. 10, line 1-18. The *servers* beneath the coordinator can be referred to as subordinates. Shaheen at Col. 10, line 1-18. Additionally, replicas contained on the various *servers* have a modification log which note when changes have been made to the replicate saved thereon. Shaheen at Col. 10, line 1-18. The coordinator *server* compares replica states of the *servers* below it in the FSG. Shaheen at Col. 10, line 1-18. The Coordinator then requests modification logs from subordinates, and those that send the logs to the Coordinator continue on in the replication. Shaheen at Col. 10, line 1-18. The Coordinator then merges the logs from all subordinates and then sends the merged log to all the subordinates. Shaheen at Col. 10, line 1-18. Therefore, as Shaheen discloses that a modification log is sent from the subordinate to the coordinator, and the subordinate contains a “first folder” and the coordinator contains a “second folder” this discloses the recitation in claim 1 “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose the recitation in claim 1 of “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 10, line 1-18.

Shaheen also discloses a new third *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. As noted above, when the coordinator receives the modification logs from the subordinates, it merges them. Shaheen at Col. 10, line 1-18. As the merged modification log is formed based on the modification logs from the subordinate *servers*, a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 1. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also

can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 1. Shaheen at Col. 10, line 1-18.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Shaheen also discloses electronically *synchronizing* the third folder and the second folder. Shaheen at Col. 10, line 1-18. As indicated above, Shaheen discloses that the Coordinator *server* requests logs from each subordinate *server* and then merges these logs. Shaheen at Col. 10, line 1-18. After merging the Coordinator sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. The Subordinates then performs the updates that were missing from its replica. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to the subordinate *servers*, and the Coordinator contains a “third folder”, and a subordinate server different from the first contains a “second folder” this therefore discloses “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the subordinate is also updated based on the changes made to the coordinator and the merged log, this also can disclose the recitation in claim 1 of “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18.

Shaheen also discloses a new second *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. As noted above, the Coordinator *server* requests modification logs from each subordinate *server*, merges them and then sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to all subordinates, wherein at least one of the subordinates will be different from the first, this therefore discloses a new second *record entry* is created in the second folder based on the new third *record entry*. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the Subordinate also is updated based on the changes made to the Coordinator and the merged log a, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 1. Shaheen at Col. 10, line 1-18.

2. Claim 2

Claim 2 of the '231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

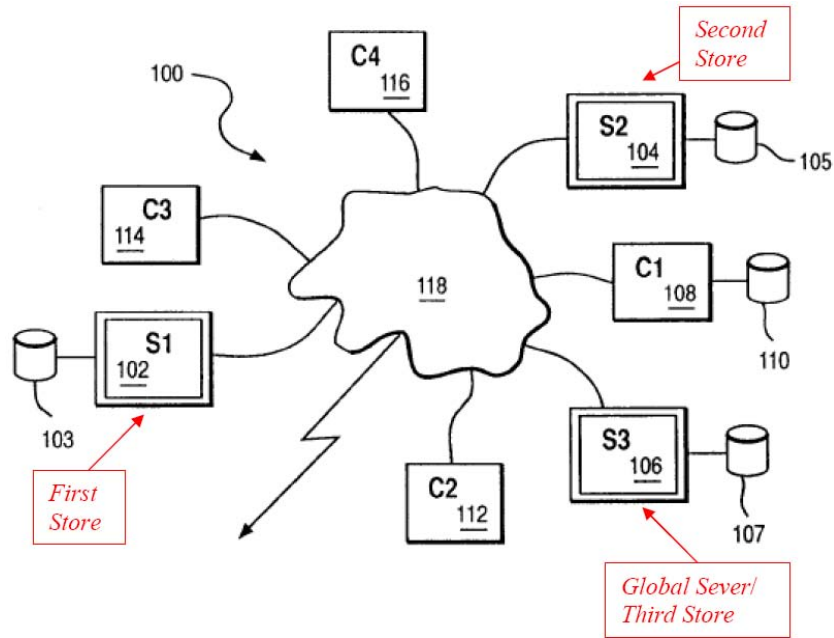
an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Shaheen.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Shaheen discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1. Specifically, Shaheen discloses a clustered or distributed system having multiple *servers*, each having non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1.



Shaheen Figure 1.

The non-volatile storage may be magnetic hard disks, optical disks or similar technology. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, Shaheen indicates that a group of these *servers* can each contain a copy of a particular fileset, where a fileset is a sub tree of a file system or an information directory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. The group of *servers* having a replica of a particular fileset are known as a fileset storage group (FSG). Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Therefore, one of the multiple *servers* with the non-volatile storage in a FSG can be considered a “*first memory store*” as recited in claim 2. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, as this *server* contains a replica of a fileset, this discloses “a first folder for storing first *record entries* in a file” according to claim 2. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6,

lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses a clustered or distributed system having multiple *servers* with non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. When a group of these *servers* contain a replica of a particular fileset, they are part of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Therefore, one *server* in the FSG can be considered a first memory, and another *server* in the FSG can be considered a second memory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As these *servers* can store filesets, a “second folder” as recited in claim 2 is disclosed. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Additionally, the fileset replica stored on the *server* in the FSG can be considered a second *record entry* in a file according to claim 2. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

Additionally, to aid in replication, each replica will have a fileset version vector. After a modification the vector is updated, and is the new fileset version vector. This modification log is then propagated through the system during replication. Such a modification log can also be considered a *record entry* in a file.

Shaheen also discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses that an FSG will contain a particular replica of a fileset. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Furthermore, if any changes take place on the *servers*, a replication process can take place wherein all the *servers* in the FSG will again contain the same replica of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Thus before and after such replication, a first folder on one *server* in the FSG will be *synchronized* with another *server* on the FSG such that their content will be the same. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Accordingly, this discloses that at a first moment in time, the contents of “the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Shaheen discloses an interface for receiving new first *record entries* in a file at the *first memory store*. Shaheen at Col. 7, lines 22-26. As disclosed in Shaheen, before replication, there is first an identification of the *servers* with replicas that are missing updates. Shaheen at Col. 7, line 22-26. Thus as some *servers* are missing updates, this indicates changes have been made to the fileset of at least one of the *servers* in the FSG. Shaheen at Col. 7, line 22-26. Thus the *server* on where the changes occurred in Shaheen can be considered as “receiving new first *record entries* in a file at the *first memory store*” according to claim 2. Shaheen at Col. 7, line 22-26. Furthermore, Figure 2 shows an “interface”, including a display, keyboard and pointer. Additionally, as the changes noted above were made to replica file sets, this therefore discloses the changes were made “at a second moment in time later than the first moment in time.” Shaheen at Col. 7, line 22-26. Also, a modification log which notes the fileset version on the *server* is changed as well when updates are made to the replica. Such changes to the modification log can also be considered “receiving new first *record entries* in a file at the *first memory store*.”

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Shaheen discloses a general synchronization module for electronically *synchronizing* the first folder and the third folder. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when replication is to occur, each of the *servers* in the FSG is given a particular priority, and a coordinator chosen. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The *servers* beneath the coordinator can be referred to as subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, replicas contained on the various *servers* have a modification log which note when changes have been made to the replicate saved thereon. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The coordinator *server* compares replica states of the *servers* below it in the FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The Coordinator then requests modification logs from subordinates, and those that send the

logs to the Coordinator continue on in the replication. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The Coordinator then merges the logs from all subordinates and then sends the merged log to all the subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Therefore, as Shaheen discloses that a modification log is sent from the subordinate to the coordinator, and the subordinate contains a “first folder” and the coordinator contains a “second folder” this discloses the recitation in claim 2 “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose the recitation in claim 2 of “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, the “general synchronization module” can be either the the Coordinator or the Subordinate. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

Shaheen also discloses a new third *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when the coordinator receives the modification logs from the subordinates, it merges them. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As the merged modification log is formed based on the modification logs from the subordinate *servers*, a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 2. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Furthermore, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 2. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Shaheen also discloses a synchronization agent for electronically *synchronizing* the third folder and the

second folder. Shaheen at Col. 10, line 1-18. As indicated above, Shaheen discloses that the Coordinator *server* requests logs from each subordinate *server* and then merges these logs. After merging the Coordinator sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. The Subordinates then performs the updates that were missing from its replica. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to the subordinate *servers*, and the Coordinator contains a “third folder”, and a subordinate server different from the first contains a “second folder” this therefore discloses “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the subordinate is also updated based on the changes made to the coordinator and the merged log, this also can disclose the recitation in claim 2 of “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Furthermore the “synchronization agent” can be either can be either the the Coordinator or the Subordinate. Shaheen at Col. 10, line 1-18.

Shaheen also discloses a new second *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. As noted above, the Coordinator *server* requests modification logs from each subordinate *server*, merges them and then sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to all subordinates, wherein at least one of the subordinates will be different from the first, this therefore discloses a new second *record entry* is created in the second folder based on the new third *record entry*. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the Subordinate also is updated based on the changes made to the Coordinator and the merged log a, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 2. Shaheen at Col. 10, line 1-18.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

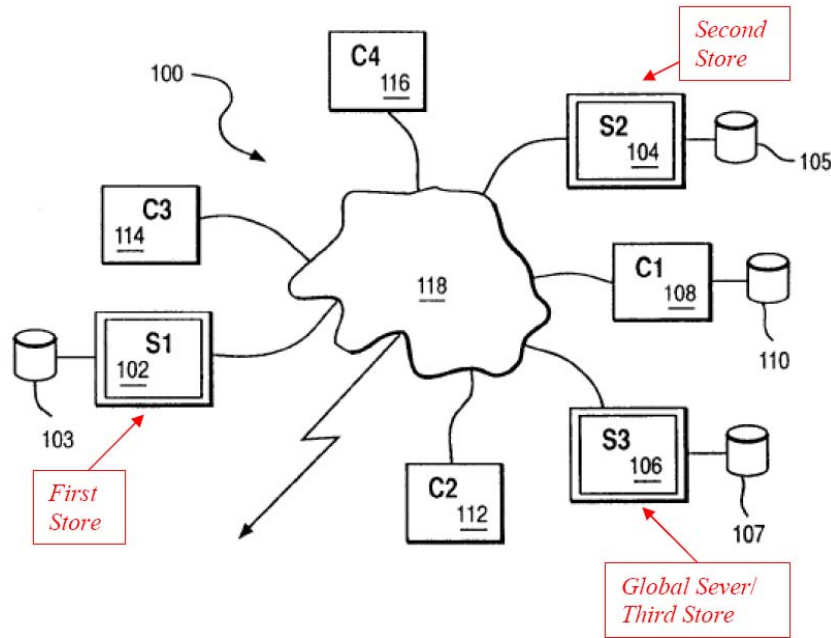
means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Shaheen.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Shaheen discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig 1. Specifically, Shaheen discloses a clustered or distributed system having multiple *servers*, each having non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1.



Shaheen Figure 1.

The non-volatile storage may be magnetic hard disks, optical disks or similar technology. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, Shaheen indicates that a group of these *servers* can each contain a copy of a particular fileset, where a fileset is a sub tree of a file system or an information directory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. The group of *servers* having a replica of a particular fileset are known as a fileset storage group (FSG). Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Therefore, one of the multiple *servers* with the non-volatile storage in a FSG can be considered a “*first memory store*” as recited in claim 3. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, as this *server* contains a replica of a fileset, this discloses “a first folder for storing first *record entries* in a file” according to claim 3. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen also discloses a second memory means coupled to the *first memory store*, the second memory means including a second folder for storing second *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6,

lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses a clustered or distributed system having multiple *servers* with non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. When a group of these *servers* contain a replica of a particular fileset, they are part of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Therefore, one *server* in the FSG can be considered a first memory, and another *server* in the FSG can be considered a second memory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As these *servers* can store filesets, a “second folder” as recited in claim 3 is disclosed. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Additionally, the fileset replica stored on the *server* in the FSG can be considered a second *record entry* in a file according to claim 1. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

Additionally, to aid in replication, each replica will have a fileset version vector. After a modification the vector is updated, and is the new fileset version vector. This modification log is then propagated through the system during replication. Such a modification log can also be considered a *record entry* in a file.

Shaheen also discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses that an FSG will contain a particular replica of a fileset. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Furthermore, if any changes take place on the *servers*, a replication process can take place wherein all the *servers* in the FSG will again contain the same replica of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Thus before and after such replication, a first folder on one *server* in the FSG will be *synchronized* with another *server* on the FSG such that their content will be the same. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Accordingly, this discloses that at a first moment in time, the contents of “the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Shaheen also discloses a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. As noted above, Shaheen discloses multiple *servers* in an FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. When a replication or update event is to occur, all the computers in the FSG are assigned a unique priority. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. A *server* in the FSG is assigned to act as a “coordinator.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. The assignment is made such that each *server* in a FSG will have only one coordinator above it. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. This coordinator computer can be considered a *global server*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. Furthermore, as this *server* contains a non-volatile memory and a replica of a fileset, this discloses a *third memory store*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Shaheen discloses a means for receiving new first *record entries* in a file at the first memory means. Shaheen at Col. 7, lines 22-26. As disclosed in Shaheen, before replication, there is first an identification of the *servers* with replicas that are missing updates. Shaheen at Col. 7, line 22-26. Thus as some *servers* are missing updates, this indicates changes have been made to the fileset of at least one of the *servers* in the FSG. Shaheen at Col. 7, line 22-26. Thus the *server* on where the changes occurred in Shaheen can be considered as “receiving new first *record entries* in a file at the *first memory store*” according to claim 3. Shaheen at Col. 7, line 22-26. Furthermore, Figure 2 shows a “means”, including a display, keyboard and pointer. Additionally, as the changes noted above were made to replica file sets, this therefore discloses the changes were made “at a second moment in time later than the first moment in time.” Shaheen at Col. 7, line 22-26. Also, a modification log which notes the fileset version on the *server* is changed as well when updates are made to the replica. Such changes to the modification log can also be considered “receiving new first *record entries* in a file at a first memory means.” Additionally, as the modification log shows updates

made to the replica they also are made “at a second moment in time later than the first moment in time.”

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Shaheen discloses a means for electronically *synchronizing* the first folder and the third folder. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when replication is to occur, each of the *servers* in the FSG is given a particular priority, and a coordinator chosen. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The *servers* beneath the coordinator can be referred to as subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, replicas contained on the various *servers* have a modification log which note when changes have been made to the replicate saved thereon. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The coordinator *server* compares replica states of the *servers* below it in the FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The Coordinator then requests modification logs from subordinates, and those that send the logs to the Coordinator continue on in the replication. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The Coordinator then merges the logs from all subordinates and then sends the merged log to all the subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Therefore, as Shaheen discloses that a modification log is sent from the subordinate to the coordinator, and the subordinate contains a “first folder” and the coordinator contains a “second folder” this discloses the recitation in claim 3 “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose the recitation in claim 3 of “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

Shaheen also discloses a new third *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when the coordinator receives the modification logs

from the subordinates, it merges them. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As the merged modification log is formed based on the modification logs from the subordinate *servers*, a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 3. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Furthermore, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 3. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Shaheen also discloses a means for electronically *synchronizing* the third folder and the second folder. Shaheen at Col. 10, line 1-18. As indicated above, Shaheen discloses that the Coordinator *server* requests logs from each subordinate *server* and then merges these logs. After merging the Coordinator sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. The Subordinates then performs the updates that were missing from its replica. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to the subordinate *servers*, and the Coordinator contains a “third folder”, and a subordinate server different from the first contains a “second folder” this therefore discloses “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the subordinate is also updated based on the changes made to the coordinator and the merged log, this also can disclose the recitation in claim 3 of “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18.

Shaheen also discloses a new second *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. As noted above, the Coordinator *server* requests modification logs from each subordinate *server*, merges them and then sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to all subordinates, wherein at least one of the subordinates will be different from the first, this therefore discloses a new second *record*

entry is created in the second folder based on the new third *record entry*. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the Subordinate also is updated based on the changes made to the Coordinator and the merged log a, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 3. Shaheen at Col. 10, line 1-18.

4. Claim 4

Claim 4 of the '231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

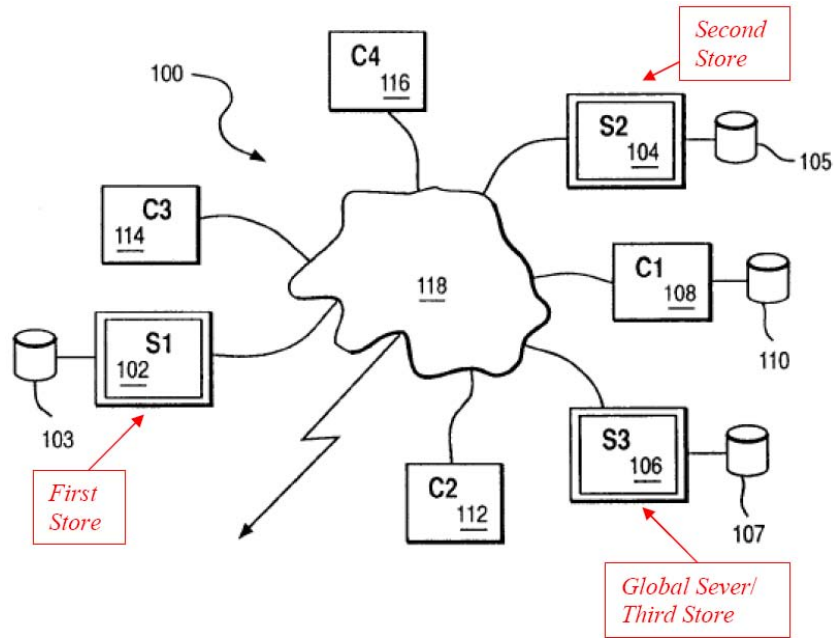
receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Shaheen.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Shaheen discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig 1. Specifically, Shaheen discloses a clustered or distributed system having multiple *servers*, each having non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1.



Shaheen Figure 1.

The non-volatile storage may be magnetic hard disks, optical disks or similar technology. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, Shaheen indicates that a group of these *servers* can each contain a copy of a particular fileset, where a fileset is a sub tree of a file system or an information directory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. The group of *servers* having a replica of a particular fileset are known as a fileset storage group (FSG). Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Therefore, one of the multiple *servers* with the non-volatile storage in a FSG can be considered a “*first memory store*” as recited in claim 4. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, as this *server* contains a replica of a fileset, this discloses “a first folder for storing first *record entries* in a file” according to claim 4. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen also discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Shaheen at Col. 5,

lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses a clustered or distributed system having multiple *servers* with non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. When a group of these *servers* contain a replica of a particular fileset, they are part of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Therefore, one *server* in the FSG can be considered a first memory, and another *server* in the FSG can be considered a second memory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As these *servers* can store filesets, a “second folder” as recited in claim 4 is disclosed. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Additionally, the fileset replica stored on the *server* in the FSG can be considered a second *record entry* in a file according to claim 4. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

Additionally, to aid in replication, each replica will have a fileset version vector. After a modification the vector is updated, and is the new fileset version vector. This modification log is then propagated through the system during replication. Such a modification log can also be considered a *record entry* in a file.

Shaheen also discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses that an FSG will contain a particular replica of a fileset. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Furthermore, if any changes take place on the *servers*, a replication process can take place wherein all the *servers* in the FSG will again contain the same replica of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Thus before and after such replication, a first folder on one *server* in the FSG will be *synchronized* with another *server* on the FSG such that their content will be the same. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Accordingly, this discloses that at a first moment in time, the contents of “the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Shaheen also discloses a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. As noted above, Shaheen discloses multiple *servers* in an FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. When a replication or update event is to occur, all the computers in the FSG are assigned a unique priority. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. A *server* in the FSG is assigned to act as a “coordinator.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. The assignment is made such that each *server* in a FSG will have only one coordinator above it. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. This coordinator computer can be considered a *global server*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. Furthermore, as this *server* contains a non-volatile memory and a replica of a fileset, this discloses a *third memory store*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58.

The fourth element of claim 4 is “receiving changes at the first memory store.” Shaheen discloses receiving changes at the *first memory store*. Shaheen at Col. 7, lines 22-26. As disclosed in Shaheen, before replication, there is first an identification of the *servers* with replicas that are missing updates. Shaheen at Col. 7, line 22-26. Thus as some *servers* are missing updates, this indicates changes have been made to the fileset of at least one of the *servers* in the FSG. Shaheen at Col. 7, line 22-26. Thus the *server* on where the changes occurred in Shaheen can be considered as “receiving changes at the *first memory store*” according to claim 4. Shaheen at Col. 7, line 22-26. Additionally a modification log which notes the fileset version on the *server* is changed as well when updates are made to the replica. Such changes to the modification log can also be considered “receiving new first changes at the *first memory store*.” Shaheen at Col. 7, line 22-26.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Shaheen discloses electronically *synchronizing* the first folder and the third folder. Shaheen at Col. 10, line 1-18. As noted above, when replication is to occur, each of the *servers* in the FSG is given a particular priority, and a coordinator chosen. Shaheen at

Col. 10, line 1-18. The *servers* beneath the coordinator can be referred to as subordinates. Shaheen at Col. 10, line 1-18. Additionally, replicas contained on the various *servers* have a modification log which note when changes have been made to the replicate saved thereon. Shaheen at Col. 10, line 1-18. The coordinator *server* compares replica states of the *servers* below it in the FSG. Shaheen at Col. 10, line 1-18. The Coordinator then requests modification logs from subordinates, and those that send the logs to the Coordinator continue on in the replication. Shaheen at Col. 10, line 1-18. The Coordinator then merges the logs from all subordinates and then sends the merged log to all the subordinates. Shaheen at Col. 10, line 1-18. Therefore, as Shaheen discloses that a modification log is sent from the subordinate to the coordinator, and the subordinate contains a “first folder” and the coordinator contains a “second folder” this discloses the recitation in claim 4 “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose the recitation in claim 4 of “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 10, line 1-18.

Shaheen also discloses the content of the changes creates a third *record entry* Shaheen at Col. 10, line 1-18. As noted above, when the coordinator receives the modification logs from the subordinates, it merges them. Shaheen at Col. 10, line 1-18. As the merged modification log is formed based on the modification logs from the subordinate *servers*, a new third *record entry* is created in the third folder as recited in claim 4. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose a new third *record entry* is created in the third folder as recited in claim 4. Shaheen at Col. 10, line 1-18.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Shaheen also discloses electronically *synchronizing* the third folder and the second folder. Shaheen at Col. 10, line 1-18. As indicated above, Shaheen discloses that the Coordinator *server* requests logs from each subordinate *server* and then merges these logs. Shaheen at Col. 10, line 1-18. After merging the

Coordinator sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. The Subordinates then performs the updates that were missing from its replica. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to the subordinate *servers*, and the Coordinator contains a “third folder”, and a subordinate server different from the first contains a “second folder” this therefore discloses “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the subordinate is also updated based on the changes made to the coordinator and the merged log, this also can disclose the recitation in claim 4 of “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18.

Shaheen also discloses a new second *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. As noted above, the Coordinator *server* requests modification logs from each subordinate *server*, merges them and then sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to all subordinates, wherein at least one of the subordinates will be different from the first, this therefore discloses a new second *record entry* is created in the second folder based on the new third *record entry*. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the Subordinate also is updated based on the changes made to the Coordinator and the merged log a, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 4. Shaheen at Col. 10, line 1-18.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

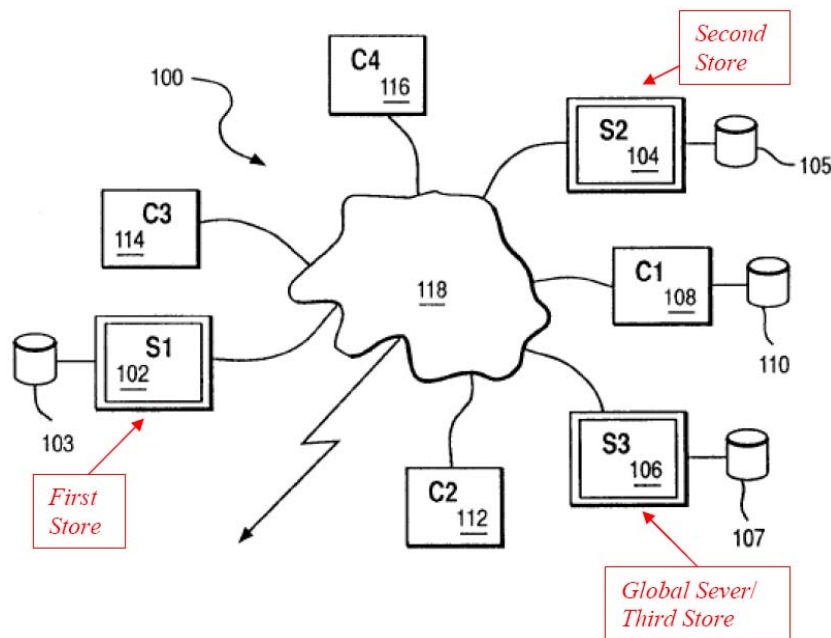
an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Shaheen.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Shaheen discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig 1. Specifically, Shaheen discloses a clustered or distributed system having multiple *servers*, each having non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1.



Shaheen Figure 1.

The non-volatile storage may be magnetic hard disks, optical disks or similar

technology. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, Shaheen indicates that a group of these *servers* can each contain a copy of a particular fileset, where a fileset is a sub tree of a file system or an information directory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. The group of *servers* having a replica of a particular fileset are known as a fileset storage group (FSG). Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Therefore, one of the multiple *servers* with the non-volatile storage in a FSG can be considered a “*first memory store*” as recited in claim 5. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, as this *server* contains a replica of a fileset, this discloses “a first folder for storing first *record entries* in a file” according to claim 5. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses a clustered or distributed system having multiple *servers* with non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. When a group of these *servers* contain a replica of a particular fileset, they are part of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Therefore, one *server* in the FSG can be considered a first memory, and another *server* in the FSG can be considered a second memory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As these *servers* can store filesets, a “second folder” as recited in claim 5 is disclosed. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Additionally, the fileset replica stored on the *server* in the FSG can be considered a second *record entry* in a file according to claim 5. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

Additionally, to aid in replication, each replica will have a fileset version vector. After a modification the vector is updated, and is the new fileset version vector. This

modification log is then propagated through the system during replication. Such a modification log can also be considered a *record entry* in a file.

Shaheen also discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses that an FSG will contain a particular replica of a fileset. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Furthermore, if any changes take place on the *servers*, a replication process can take place wherein all the *servers* in the FSG will again contain the same replica of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Thus before and after such replication, a first folder on one *server* in the FSG will be *synchronized* with another *server* on the FSG such that their content will be the same. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Accordingly, this discloses that at a first moment in time, the contents of “the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Shaheen also discloses a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. As noted above, Shaheen discloses multiple *servers* in an FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. When a replication or update event is to occur, all the computers in the FSG are assigned a unique priority. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. A *server* in the FSG is assigned to act as a “coordinator.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. The assignment is made such that each *server* in a FSG will have only one coordinator above it. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. This coordinator computer can be considered a *global server*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. Furthermore, as this *server* contains a non-volatile memory and a replica of a fileset, this discloses a *third memory store*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Shaheen discloses an interface for receiving changes at the *first memory store*. Shaheen at Col. 7, lines 22-26. As disclosed in Shaheen, before replication, there is first an identification of the *servers* with replicas that are missing updates. Shaheen at Col. 7, line 22-26. Thus as some *servers* are missing updates, this indicates changes have been made to the fileset of at least one of the *servers* in the FSG. Shaheen at Col. 7, line 22-26. Thus the *server* on where the changes occurred in Shaheen can be considered as “receiving new first *record entries* in a file at the *first memory store*” according to claim 5. Shaheen at Col. 7, line 22-26. Furthermore, Figure 2 shows an “interface”, including a display, keyboard and pointer. Additionally, as the changes noted above were made to replica file sets, this therefore discloses the changes were made “at a second moment in time later than the first moment in time.” Shaheen at Col. 7, line 22-26. Also, a modification log which notes the fileset version on the *server* is changed as well when updates are made to the replica. Such changes to the modification log can also be considered “receiving changes at the *first memory store*.”

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Shaheen discloses a general synchronization module for electronically *synchronizing* the first folder and the third folder. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when replication is to occur, each of the *servers* in the FSG is given a particular priority, and a coordinator chosen. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The *servers* beneath the coordinator can be referred to as subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, replicas contained on the various *servers* have a modification log which note when changes have been made to the replicate saved thereon. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The coordinator *server* compares replica states of the *servers* below it in the FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The Coordinator then requests modification logs from subordinates, and those that send the logs to the Coordinator continue on in the replication. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58,

Col. 10, line 1-18. The Coordinator then merges the logs from all subordinates and then sends the merged log to all the subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Therefore, as Shaheen discloses that a modification log is sent from the subordinate to the coordinator, and the subordinate contains a “first folder” and the coordinator contains a “second folder” this discloses the recitation in claim 5 “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose the recitation in claim 5 of “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, the “general synchronization module” can be either the Coordinator or the Subordinate. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

Shaheen also discloses a new third *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when the coordinator receives the modification logs from the subordinates, it merges them. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As the merged modification log is formed based on the modification logs from the subordinate *servers*, a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 5. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Furthermore, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 5. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Shaheen also discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Shaheen at Col. 10, line 1-18. As indicated above, Shaheen discloses that the Coordinator *server* requests logs from each subordinate *server* and then merges these

logs. After merging the Coordinator sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. The Subordinates then performs the updates that were missing from its replica. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to the subordinate *servers*, and the Coordinator contains a “third folder”, and a subordinate server different from the first contains a “second folder” this therefore discloses “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the subordinate is also updated based on the changes made to the coordinator and the merged log, this also can disclose the recitation in claim 5 of “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Furthermore the “synchronization agent” can be either the Coordinator or the Subordinate. Shaheen at Col. 10, line 1-18.

Shaheen also discloses a new second *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. As noted above, the Coordinator *server* requests modification logs from each subordinate *server*, merges them and then sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to all subordinates, wherein at least one of the subordinates will be different from the first, this therefore discloses a new second *record entry* is created in the second folder based on the new third *record entry*. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the Subordinate also is updated based on the changes made to the Coordinator and the merged log a, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 5. Shaheen at Col. 10, line 1-18.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

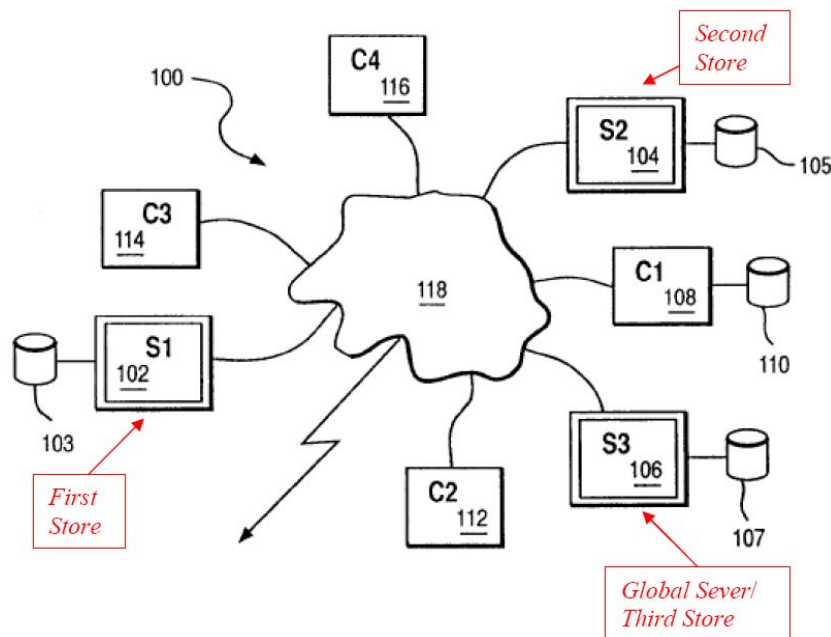
means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Shaheen.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Shaheen discloses providing a *first memory store* including a first folder for storing first *record entries* in a file Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig 1. Specifically, Shaheen discloses a clustered or distributed system having multiple *servers*, each having non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45 and Fig. 1.



Shaheen Figure 1.

The non-volatile storage may be magnetic hard disks, optical disks or similar technology. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, Shaheen indicates that a group of these *servers* can each contain a copy of a particular fileset.

where a fileset is a sub tree of a file system or an information directory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. The group of *servers* having a replica of a particular fileset are known as a fileset storage group (FSG). Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Therefore, one of the multiple *servers* with the non-volatile storage in a FSG can be considered a “*first memory store*” as recited in claim 6. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45. Additionally, as this *server* contains a replica of a fileset, this discloses “a first folder for storing first *record entries* in a file” according to claim 6. Shaheen at Col. 5, lines 13-19, Col. 6, lines 38-45.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses a clustered or distributed system having multiple *servers* with non-volatile storage. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. When a group of these *servers* contain a replica of a particular fileset, they are part of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Therefore, one *server* in the FSG can be considered a first memory, and another *server* in the FSG can be considered a second memory. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As these *servers* can store filesets, a “second folder” as recited in claim 6 is disclosed. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Additionally, the fileset replica stored on the *server* in the FSG can be considered a second *record entry* in a file according to claim 6. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

Additionally, to aid in replication, each replica will have a fileset version vector. After a modification the vector is updated, and is the new fileset version vector. This modification log is then propagated through the system during replication. Such a modification log can also be considered a *record entry* in a file.

Shaheen also discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. As discussed above, Shaheen discloses that an FSG will contain a particular replica of a fileset. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Furthermore, if any changes take place on the *servers*, a replication process can take place wherein all the *servers* in the FSG will again contain the same replica of an FSG. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Thus before and after such replication, a first folder on one *server* in the FSG will be *synchronized* with another *server* on the FSG such that their content will be the same. Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26. Accordingly, this discloses that at a first moment in time, the contents of “the second folder is consistent with the content of the first folder at the first moment in time.” Shaheen at Col. 5, lines 13-19, Col. 6, lines 37-45, Col. 7, lines 22-26.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Shaheen also discloses a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. As noted above, Shaheen discloses multiple *servers* in an FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. When a replication or update event is to occur, all the computers in the FSG are assigned a unique priority. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. A *server* in the FSG is assigned to act as a “coordinator.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. The assignment is made such that each *server* in a FSG will have only one coordinator above it. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. This coordinator computer can be considered a *global server*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58. Furthermore, as this *server* contains a non-volatile memory and a replica of a fileset, this discloses a *third memory store*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Shaheen discloses means for receiving changes at the *first memory store*. Shaheen at Col. 7, lines 22-26. As disclosed in Shaheen, before replication, there is first

an identification of the *servers* with replicas that are missing updates. Shaheen at Col. 7, line 22-26. Thus as some *servers* are missing updates, this indicates changes have been made to the fileset of at least one of the *servers* in the FSG. Shaheen at Col. 7, line 22-26. Thus the *server* on where the changes occurred in Shaheen can be considered as “receiving changes at the *first memory store*” according to claim 6. Shaheen at Col. 7, line 22-26. Additionally a modification log which notes the fileset version on the *server* is changed as well when updates are made to the replica. Such changes to the modification log can also be considered “receiving new first changes at the *first memory store*.” Shaheen at Col. 7, line 22-26.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Shaheen discloses a means for electronically *synchronizing* the first folder and the third folder. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when replication is to occur, each of the *servers* in the FSG is given a particular priority, and a coordinator chosen. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The *servers* beneath the coordinator can be referred to as subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Additionally, replicas contained on the various *servers* have a modification log which note when changes have been made to the replicate saved thereon. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The coordinator *server* compares replica states of the *servers* below it in the FSG. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The Coordinator then requests modification logs from subordinates, and those that send the logs to the Coordinator continue on in the replication. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. The Coordinator then merges the logs from all subordinates and then sends the merged log to all the subordinates. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Therefore, as Shaheen discloses that a modification log is sent from the subordinate to the coordinator, and the subordinate contains a “first folder” and the coordinator contains a “second folder” this discloses the recitation in claim 6 “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-

18. Additionally, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose the recitation in claim 6 of “electronically *synchronizing* the first folder and the third folder.” Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

Shaheen also discloses a new third *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As noted above, when the coordinator receives the modification logs from the subordinates, it merges them. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. As the merged modification log is formed based on the modification logs from the subordinate *servers*, a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 6. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18. Furthermore, as the replica contained in the coordinator also is updated based on the changes made to the subordinate as noted in the modification log, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 6. Shaheen at Col. 5, lines 13-19, Col. 8, lines 11-20, 51-58, Col. 10, line 1-18.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Shaheen also discloses a means for electronically *synchronizing* the third folder and the second folder. Shaheen at Col. 10, line 1-18. As indicated above, Shaheen discloses that the Coordinator *server* requests logs from each subordinate *server* and then merges these logs. After merging the Coordinator sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. The Subordinates then performs the updates that were missing from its replica. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to the subordinate *servers*, and the Coordinator contains a “third folder”, and a subordinate server different from the first contains a “second folder” this therefore discloses “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18. Additionally, as the replica contained in the subordinate is also updated based on the changes made to the coordinator and the merged log, this also can disclose the recitation

in claim 6 of “electronically *synchronizing* the third folder and the second folder.” Shaheen at Col. 10, line 1-18.

Shaheen also discloses a new second *record entry* is created in the third folder based on the new first *record entries*. Shaheen at Col. 10, line 1-18. As noted above, the Coordinator *server* requests modification logs from each subordinate *server*, merges them and then sends the merged log to all subordinates. Shaheen at Col. 10, line 1-18. As the merged log is sent from the Coordinator to all subordinates, wherein at least one of the subordinates will be different from the first, this therefore discloses a new second *record entry* is created in the second folder based on the new third *record entry*. Shaheen at Col. 10, line 1-18. Furthermore, as the replica contained in the Subordinate also is updated based on the changes made to the Coordinator and the merged log a, this also can disclose a new third *record entry* is created in the third folder based on the new first *record entries* as recited in claim 6. Shaheen at Col. 10, line 1-18.

H. Claims 1-6 are Anticipated by Wright Under 35 U.S.C. § 102

**Please see attached Exhibit CC-G
presenting claim charts for
comparison of the Wright with claims
1-6 of the ‘231 patent.**

Requester respectfully submits that claims 1-6 of the ‘231 patent are anticipated by Wright under 35 U.S.C. § 102(b). A claim chart applying Wright to these claims is submitted herewith as Exhibit CC-G.

1. Claim 1

Claim 1 of the ‘231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second

folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

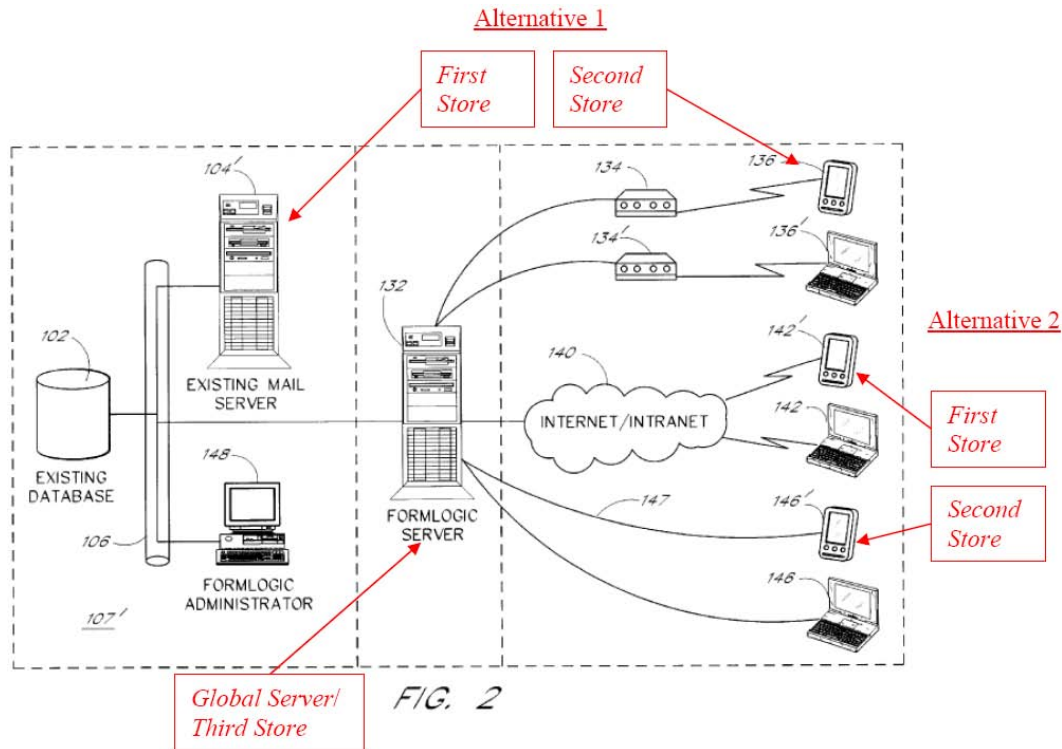
receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Wright.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Wright discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. In particular, Wright discloses a FormLogic (FL) client/*server* system. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL client includes an FL engine which allows FL applications to execute on a variety of handheld devices, such as personal digital assistants (PDA) as well as other portable devices such as laptops. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Therefore the FL client of Wright discloses a “*first memory store*” as recited in claim 1. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Alternatively, a mail *server* 104’ in Wright can be considered a “*first memory store*” which also can be connected with the FL clients via the internet. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.



Wright Fig. 2.

Wright also discloses that FL clients “need to be able to store and access information.” Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL engine also incorporates a full local database for manipulation and collection of data. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The client can have items stored such as email, a data, or work orders. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Such data or information discloses the “first *record entries* in a file” limitation as recited in claim 1. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Wright at Col. 6, lines 10-21; Fig. 2. As discussed above, Wright discloses a FL client/server system.

Wright at Col. 6, lines 10-21; Fig. 2. Clients, such as a PDA or laptop are connected to the FL *server*. Wright at Col. 6, lines 10-21; Fig. 2. One client, or a mail *server*, connected with the *server* can be considered a “*first memory store*” while another client on the network can be considered a “*second memory store*.” Wright at Col. 6, lines 10-21; Fig. 2. As they are connected by Local area network, or by the internet, they are then “coupled” as recited in claim 1. Wright at Col. 6, lines 10-21; Fig. 2.

Additionally, Wright discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Wright at Col. 6, lines 10-21; Fig. 2. Wright discloses that an application programming interface “allows developers to easily extend the FL *server*...to their data sources.” Wright at Col. 6, lines 10-21; Fig. 2. Additionally, data can be transferred between the data sources and the PDA’s. As a result of such a “session” there is “*synchronizing*” between the data source and the clients. Wright at Col. 6, lines 10-21; Fig. 2. Thus, Wright discloses a “second folder being *synchronized* with a first folder at a first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2. Additionally, after such synchronization, or before any changes are made to the data source or client database, the contents of the data source and FL client would be the same thus disclosing the limitation in claim 1 “a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Wright discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. In particular, and as discussed above, Wright discloses a client/*server* system. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* provides the ability to link hardware devices running the FL engine to access existing enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server*, acts as a “gateway” between FL clients and enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* is a multi-tier client/*server* model in that it creates an intermediate *server* between a client and a traditional, or original *server*. Wright at Col. 6, lines 22-23; Fig. 2. Thus the FL *server* and data sources of Wright can disclose a *global server* as

recited in claim 1. Wright at Col. 6, lines 22-23; Fig. 2. Additionally, as it is a “gateway” and *server* it will transmit data or information between clients and data sources. Wright at Col. 6, lines 22-23; Fig. 2. Therefore, this discloses a “*third memory store* having a third folder for storing third *record entries* in a file.” Wright at Col. 6, lines 22-23; Fig. 2.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Wright discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Wright at Col. 5, lines 46-52. Wright discloses that because FL clients do not maintain persistent connections with the FL *server*, they must be able to store and access information while not connected to the host database or other data source. Wright at Col. 5, lines 46-52. Therefore, the FL engine included in the FL *server* incorporates a full local database implementation that allows data to be manipulated and collected by the FL client while not connected to the FL *server*. Wright at Col. 5, lines 46-52. Since the data on the client can be manipulated and collected, this discloses the above mentioned limitation. Wright at Col. 5, lines 46-52.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Wright discloses electronically *synchronizing* the first folder and the third folder, such that a new third *record entry* is created in the third folder based on the new first *record entries*. Wright at Col. 5, lines 63-66. Specifically Wright discloses that the FL *server* can connect with the FL client database, and retrieve data that has been collected by the client since the last connection. Wright at Col. 5, lines 63-66. Furthermore, as the FL client can be considered to disclose a “first folder” as recited in claim 1, and the FL *server* can be considered to disclose a “third folder” as recited in claim 1, Wright therefore discloses “electronically *synchronizing* the first folder and the third folder.” Wright at Col. 5, lines 63-66. Alternatively, the *server* can retrieve emails and other data from the email *server*, which can be considered a first memory having a first folder, as discussed above. Wright at Col. 5, lines 63-66. Additionally because the data is retrieved from the FL client to the FL *server*, this also discloses “a new third entry

is created in the third folder based on the new first *record entries*.” Wright at Col. 5, lines 63-66.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Wright also discloses electronically *synchronizing* the third folder and the second folder, such that a new second *record entry* is created in the second folder based on the new third *record entry*. Wright at Col. 5, lines 52-59. In particular, Wright discloses that the FL *server* can query the database on the client in order to add data to the client database, or remove data from the client database in order to make updates to both the client and *server* database to reflect changes that have happened on both sides since the last connection. Wright at Col. 5, lines 52-59. Wright states as a result “[t]hus, a synchronization of the two databases is performed.” Wright at Col. 5, lines 52-59. Therefore as the “third folder” as recited in claim 1 is included on the *server* side, and the “second folder” is included in one of the clients in the network different from the first, this therefore discloses “*synchronizing* the third folder and the second folder.” Wright at Col. 5, lines 52-59. Furthermore as Wright discloses that the *server* is querying the client in order to add data to the client database the new data on the client is created “based on the new third *record entry*”, wherein the third record were updates from the first client as discussed above. Wright at Col. 5, lines 52-59.

2. Claim 2

Claim 2 of the ‘231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

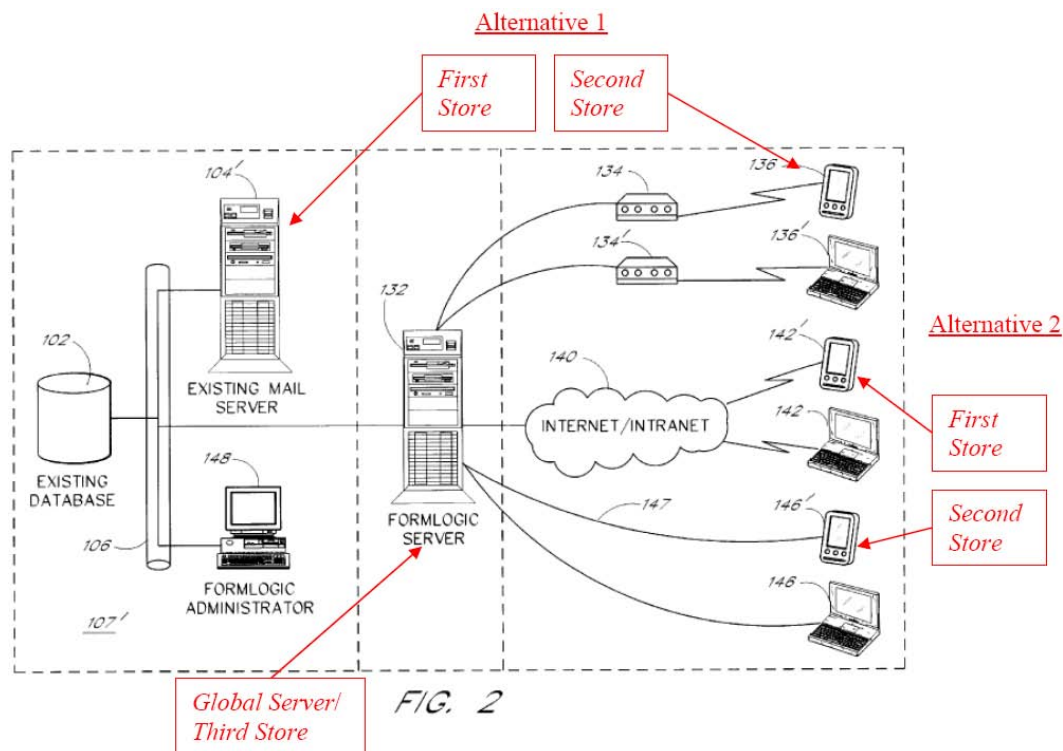
an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Wright.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Wright discloses a *first memory store* including a first folder for storing first *record entries* in a file. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. In particular, Wright discloses a FormLogic (FL) client/*server* system. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL client includes an FL engine which allows FL applications to execute on a variety of handheld devices, such as personal digital assistants (PDA) as well as other portable devices such as laptops. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Therefore the FL client of Wright discloses a “*first memory store*” as recited in claim 2. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Alternatively, a mail *server* 104’ in Wright can be considered a “*first memory store*” which also can be connected with the FL clients via the internet. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.



Wright Fig. 2.

Wright also discloses that FL clients “need to be able to store and access information.” Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL engine also incorporates a full local database for manipulation and collection of data. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The client can have items stored such as email, a data, or work orders. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Such data or information discloses the “first *record entries* in a file” limitation as recited in claim 2. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Wright at Col. 6, lines 10-21; Fig. 2. As discussed above, Wright discloses a FL client/server system. Wright at Col. 6, lines 10-

21; Fig. 2. Clients, such as a PDA or laptop are connected to the FL *server*. Wright at Col. 6, lines 10-21; Fig. 2. One client, or a mail *server*, connected with the *server* can be considered a “*first memory store*” while another client on the network can be considered a “*second memory store*.” Wright at Col. 6, lines 10-21; Fig. 2. As they are connected by Local area network, or by the internet, they are then “coupled” as recited in claim 2. Wright at Col. 6, lines 10-21; Fig. 2.

Additionally, Wright discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Wright at Col. 6, lines 10-21; Fig. 2. Wright discloses that an application programming interface allows developers to easily extend the FL *server*...to their data sources. Wright at Col. 6, lines 10-21; Fig. 2. Additionally, data can be transferred between the data sources and the PDA’s. As a result of such a “session” there is “*synchronizing*” between the data source and the clients. Wright at Col. 6, lines 10-21; Fig. 2. Thus, Wright discloses a “second folder being *synchronized* with a first folder at a first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2. Additionally, after such synchronization, or before any changes are made to the data source or client database, the contents of the data source and FL client would be the same thus disclosing the limitation in claim 2 “a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Wright discloses an interface for receiving new *record entries* in a file at the *first memory store*. Wright at Col. 5, lines 46-52. Wright discloses that because FL clients do not maintain persistent connections with the FL *server*, they must be able to store and access information while not connected to the host database or other data source. Wright at Col. 5, lines 46-52. Therefore, the FL engine included in the FL *server* incorporates a full local database implementation that allows data to be manipulated and collected by the FL client while not connected to the FL *server*. Wright at Col. 5, lines 46-52. Since the data on the client can be manipulated and collected, this discloses the above mentioned limitation. Wright at Col. 5, lines 46-52. Furthermore, as can be seen in Figure 2 of Wright, the display and input buttons on the PDA’s and laptops

can be considered an “interface” as recited in claim 2.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Wright discloses a general synchronization module for electronically *synchronizing* the first folder and the third folder such that a new third *record entry* is created in the third folder based on the new first *record entries*. Wright at Col. 5, lines 63-66; Fig. 2. Specifically Wright discloses that the FL *server* can connect with the FL client database, and retrieve data that has been collected by the client since the last connection. Wright at Col. 5, lines 63-66; Fig. 2. Furthermore, as the FL client can be considered to disclose a “first folder” as recited in claim 2, and the FL *server* can be considered to disclose a “third folder” as recited in claim 2, Wright therefore discloses “electronically *synchronizing* the first folder and the third folder.” Wright at Col. 5, lines 63-66; Fig. 2. Alternatively, the *server* can retrieve emails and other data from the email *server*, which can be considered a first memory having a first folder, as discussed above. Wright at Col. 5, lines 63-66; Fig. 2. Additionally because the data is retrieved from the FL client to the FL *server*, this also discloses “a new third entry is created in the third folder based on the new first *record entries*.” Wright at Col. 5, lines 63-66; Fig. 2. Furthermore the “synchronization module” can be the FL *server*, FL client or the mail *server*. Wright at Col. 5, lines 63-66; Fig. 2.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Wright also discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder such that a new second *record entry* is created in the second folder based on the new third *record entry*. Wright at Col. 5, lines 52-59. In particular, Wright discloses that the FL *server* can query the database on the client in order to add data to the client database, or remove data from the client database in order to make updates to both the client and *server* database to reflect changes that have happened on both sides since the last connection. Wright at Col. 5, lines 52-59. Wright states as a result “[t]hus, a synchronization of the two databases is performed.” Wright at Col. 5, lines 52-59.

Therefore as the “third folder” as recited in claim 2 is included on the *server* side, and the “second folder” is included in one of the clients in the network different from the first, this therefore discloses “*synchronizing* the third folder and the second folder.” Wright at Col. 5, lines 52-59. Furthermore as Wright discloses that the *server* is querying the client in order to add data to the client database the new data on the client is created “based on the new third *record entry*”, wherein the third record were updates from the first client as discussed above. Wright at Col. 5, lines 52-59. Furthermore the “synchronization agent” can be the FL *server*, FL client or the mail *server*. Wright at Col. 5, lines 52-59.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

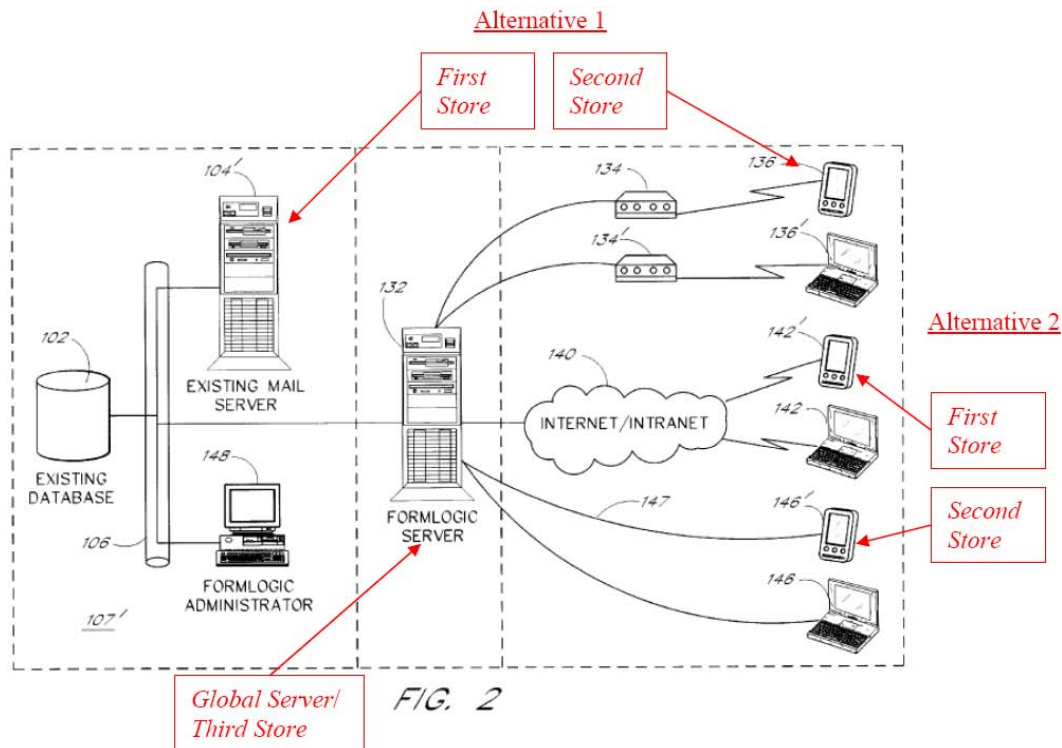
means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Wright.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Wright discloses a first memory means including a first folder for storing first *record entries* in a file. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. In particular, Wright discloses a FormLogic (FL) client/*server* system. Wright at

Col. 5, lines 16-24, 46-52; Fig. 2. The FL client includes an FL engine which allows FL applications to execute on a variety of handheld devices, such as personal digital assistants (PDA) as well as other portable devices such as laptops. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Therefore the FL client of Wright discloses a “first memory means” as recited in claim 3. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Alternatively, a mail server 104’ in Wright can be considered a “first memory means” which also can be connected with the FL clients via the internet. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.



Wright Fig. 2.

Wright also discloses that FL clients “need to be able to store and access information.” Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL engine also incorporates a full local database for manipulation and collection of data. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The client can have items stored such as email, a data, or work orders. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Such data or information discloses the “first record entries in a file” limitation as recited in claim 3. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright discloses a second memory means coupled to the *first memory store*, the second memory means including a second folder for storing second *record entries* in a file. Wright at Col. 6, lines 10-21; Fig. 2. As discussed above, Wright discloses a FL client/server system. Wright at Col. 6, lines 10-21; Fig. 2. Clients, such as a PDA or laptop are connected to the FL *server*. Wright at Col. 6, lines 10-21; Fig. 2. One client, or a mail *server*, connected with the *server* can be considered a “first memory means” while another client on the network can be considered a “second memory means.” Wright at Col. 6, lines 10-21; Fig. 2. As they are connected by Local area network, or by the internet, they are then “coupled” as recited in claim 3. Wright at Col. 6, lines 10-21; Fig. 2.

Additionally, Wright discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Wright at Col. 6, lines 10-21; Fig. 2. Wright discloses that an application programming interface allows developers to easily extend the FL *server*...to their data sources. Wright at Col. 6, lines 10-21; Fig. 2. Additionally, data can be transferred between the data sources and the PDA's. As a result of such a “session” there is “*synchronizing*” between the data source and the clients. Wright at Col. 6, lines 10-21; Fig. 2. Thus, Wright discloses a “second folder being *synchronized* with a first folder at a first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2. Additionally, after such synchronization, or before any changes are made to the data source or client database, the contents of the data source and FL client would be the same thus disclosing the limitation in claim 3 “a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Wright discloses providing a *global server* including a *third memory store* having a third folder

for storing third *record entries* in a file. In particular, and as discussed above, Wright discloses a client/server system. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* provides the ability to link hardware devices running the FL engine to access existing enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server*, acts as a “gateway” between FL clients and enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* is a multi-tier client/server model in that it creates an intermediate *server* between a client and a traditional, or original *server*. Wright at Col. 6, lines 22-23; Fig. 2. Thus the FL *server* and data sources of Wright can disclose a *global server* as recited in claim 3. Wright at Col. 6, lines 22-23; Fig. 2. Additionally, as it is a “gateway” and *server* it will transmit data or information between clients and data sources. Wright at Col. 6, lines 22-23; Fig. 2. Therefore, this discloses a “*third memory store* having a third folder for storing third *record entries* in a file.” Wright at Col. 6, lines 22-23; Fig. 2.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Wright discloses a means for receiving new first *record entries* in a file. Wright at Col. 5, lines 46-52. Wright discloses that because FL clients do not maintain persistent connections with the FL *server*, they must be able to store and access information while not connected to the host database or other data source. Wright at Col. 5, lines 46-52. Therefore, the FL engine included in the FL *server* incorporates a full local database implementation that allows data to be manipulated and collected by the FL client while not connected to the FL *server*. Wright at Col. 5, lines 46-52. Since the data on the client can be manipulated and collected, this discloses the above mentioned limitation. Wright at Col. 5, lines 46-52.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Wright discloses a means for electronically *synchronizing* the first folder and the third folder such that a new third *record entry* is created in the third folder based on the new first *record entries*. Wright at Col. 5, lines 63-66; Fig. 2. Specifically Wright discloses that the FL *server* can connect with the FL client database, and retrieve data that has been collected by the client since the last connection. Wright at Col. 5, lines 63-66; Fig. 2. Furthermore, as the FL client can be

considered to disclose a “first folder” as recited in claim 3, and the FL *server* can be considered to disclose a “third folder” as recited in claim 3, Wright therefore discloses “electronically *synchronizing* the first folder and the third folder.” Wright at Col. 5, lines 63-66; Fig. 2. Alternatively, the *server* can retrieve emails and other data from the email *server*, which can be considered a first memory having a first folder, as discussed above. Wright at Col. 5, lines 63-66; Fig. 2. Additionally because the data is retrieved from the FL client to the FL *server*, this also discloses “a new third entry is created in the third folder based on the new first *record entries*.” Wright at Col. 5, lines 63-66; Fig. 2.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Wright also discloses a means for electronically *synchronizing* the third folder and the second folder such that a new second *record entry* is created in the second folder based on the new third *record entry*. Wright at Col. 5, lines 52-59. In particular, Wright discloses that the FL *server* can query the database on the client in order to add data to the client database, or remove data from the client database in order to make updates to both the client and *server* database to reflect changes that have happened on both sides since the last connection. Wright at Col. 5, lines 52-59. Wright states as a result “[t]hus, a synchronization of the two databases is performed.” Wright at Col. 5, lines 52-59. Therefore as the “third folder” as recited in claim 3 is included on the *server* side, and the “second folder” is included in one of the clients in the network different from the first, this therefore discloses “*synchronizing* the third folder and the second folder.” Wright at Col. 5, lines 52-59. Furthermore as Wright discloses that the *server* is querying the client in order to add data to the client database the new data on the client is created “based on the new third *record entry*”, wherein the third record were updates from the first client as discussed above. Wright at Col. 5, lines 52-59.

4. Claim 4

Claim 4 of the ‘231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

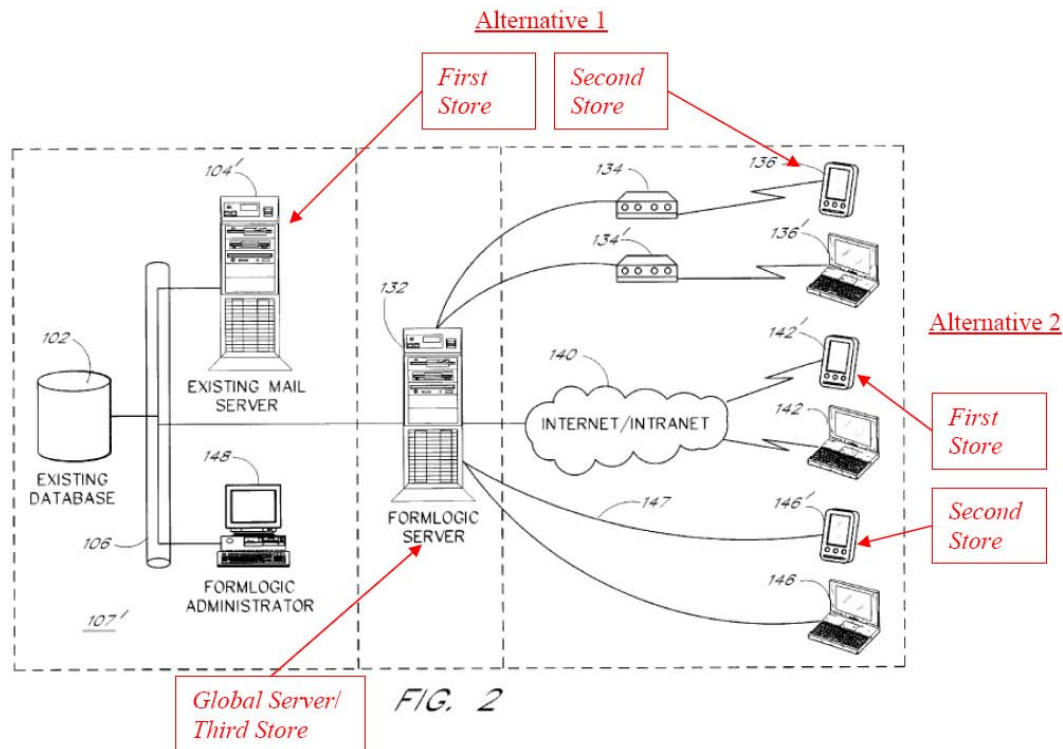
receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Wright.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Wright discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. In particular, Wright discloses a FormLogic (FL) client/*server* system. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL client includes an FL engine which allows FL applications to execute on a variety of handheld devices, such as personal digital assistants (PDA) as well as other portable devices such as laptops. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Therefore the FL client of Wright discloses a “*first memory store*” as recited in claim 4. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Alternatively, a mail *server* 104’ in Wright can be considered a “*first memory store*” which also can be connected with the FL clients via the internet. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.



Wright Fig. 2.

Wright also discloses that FL clients “need to be able to store and access information.” Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL engine also incorporates a full local database for manipulation and collection of data. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The client can have items stored such as email, a data, or work orders. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Such data or information discloses the “first *record entries* in a file” limitation as recited in claim 4. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright discloses providing a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Wright at Col. 6, lines 10-21; Fig. 2. As discussed above, Wright discloses a FL client/server system.

Wright at Col. 6, lines 10-21; Fig. 2. Clients, such as a PDA or laptop are connected to the FL *server*. Wright at Col. 6, lines 10-21; Fig. 2. One client, or a mail *server*, connected with the *server* can be considered a “*first memory store*” while another client on the network can be considered a “*second memory store*.” Wright at Col. 6, lines 10-21; Fig. 2. As they are connected by Local area network, or by the internet, they are then “coupled” as recited in claim 4. Wright at Col. 6, lines 10-21; Fig. 2.

Additionally, Wright discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Wright at Col. 6, lines 10-21; Fig. 2. Wright discloses that an application programming interface allows developers to easily extend the FL *server*...to their data sources. Wright at Col. 6, lines 10-21; Fig. 2. Additionally, data can be transferred between the data sources and the PDA's. As a result of such a “session” there is “*synchronizing*” between the data source and the clients. Wright at Col. 6, lines 10-21; Fig. 2. Thus, Wright discloses a “second folder being *synchronized* with a first folder at a first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2. Additionally, after such synchronization, or before any changes are made to the data source or client database, the contents of the data source and FL client would be the same thus disclosing the limitation in claim 4 “a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Wright discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. In particular, and as discussed above, Wright discloses a client/*server* system. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* provides the ability to link hardware devices running the FL engine to access existing enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server*, acts as a “gateway” between FL clients and enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* is a multi-tier client/*server* model in that it creates an intermediate *server* between a client and a traditional, or original *server*. Wright at Col. 6, lines 22-23; Fig. 2. Thus the FL *server* and data sources of Wright can disclose a *global server* as

recited in claim 4 Wright at Col. 6, lines 22-23; Fig. 2. Additionally, as it is a “gateway” and *server* it will transmit data or information between clients and data sources. Wright at Col. 6, lines 22-23; Fig. 2. Therefore, this discloses a “*third memory store* having a third folder for storing third *record entries* in a file.” Wright at Col. 6, lines 22-23; Fig. 2.

The fourth element of claim 4 is “receiving changes at the first memory store.” Wright discloses a means for receiving changes at the *first memory store*. Wright at Col. 5, lines 46-52. Wright discloses that because FL clients do not maintain persistent connections with the FL *server*, they must be able to store and access information while not connected to the host database or other data source. Wright at Col. 5, lines 46-52. Therefore, the FL engine included in the FL *server* incorporates a full local database implementation that allows data to be manipulated and collected by the FL client while not connected to the FL *server*. Wright at Col. 5, lines 46-52. Since the data on the client can be manipulated and collected, this discloses the above mentioned limitation. Wright at Col. 5, lines 46-52.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Wright discloses a means for electronically *synchronizing* the first folder and the third folder such that the content of the changes creates a third *record entry*. Wright at Col. 5, lines 63-66; Fig. 2. Specifically Wright discloses that the FL *server* can connect with the FL client database, and retrieve data that has been collected by the client since the last connection. Wright at Col. 5, lines 63-66; Fig. 2. Furthermore, as the FL client can be considered to disclose a “first folder” as recited in claim 4, and the FL *server* can be considered to disclose a “third folder” as recited in claim 4, Wright therefore discloses “electronically *synchronizing* the first folder and the third folder.” Wright at Col. 5, lines 63-66; Fig. 2. Alternatively, the *server* can retrieve emails and other data from the email *server*, which can be considered a first memory having a first folder, as discussed above. Wright at Col. 5, lines 63-66; Fig. 2. Additionally because the data is retrieved from the FL client to the FL *server*, this also discloses “content of the changes creates a third *record entry*.” Wright at Col. 5, lines 63-66; Fig. 2.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Wright also discloses a means for electronically *synchronizing* the third folder and the second folder such that a new second *record entry* is created in the second folder based on the new third *record entry*. Wright at Col. 5, lines 52-59. In particular, Wright discloses that the FL *server* can query the database on the client in order to add data to the client database, or remove data from the client database in order to make updates to both the client and *server* database to reflect changes that have happened on both sides since the last connection. Wright at Col. 5, lines 52-59. Wright states as a result “[t]hus, a synchronization of the two databases is performed.” Wright at Col. 5, lines 52-59. Therefore as the “third folder” as recited in claim 4 is included on the *server* side, and the “second folder” is included in one of the clients in the network different from the first, this therefore discloses “*synchronizing* the third folder and the second folder.” Wright at Col. 5, lines 52-59. Furthermore as Wright discloses that the *server* is querying the client in order to add data to the client database the new data on the client is created “based on the new third *record entry*”, wherein the third record were updates from the first client as discussed above. Wright at Col. 5, lines 52-59.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

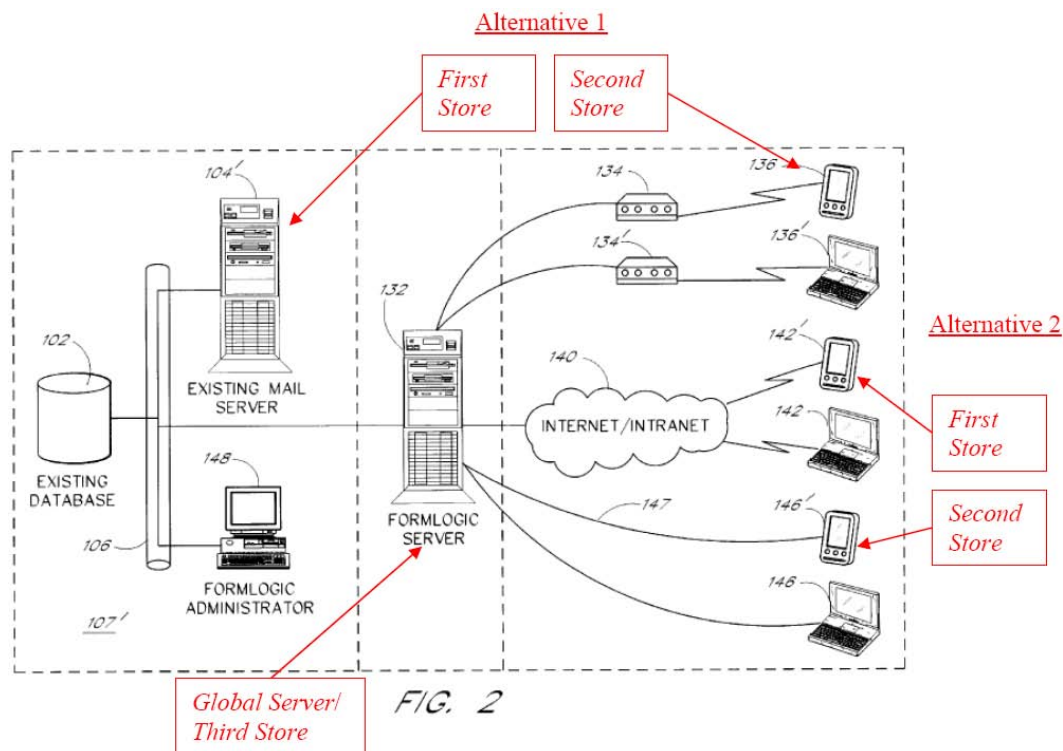
an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Wright.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Wright discloses a *first memory store* including a first folder for storing first *record entries* in a file. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. In particular, Wright discloses a FormLogic (FL) client/*server* system. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL client includes an FL engine which allows FL applications to execute on a variety of handheld devices, such as personal digital assistants (PDA) as well as other portable devices such as laptops. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Therefore the FL client of Wright discloses a “*first memory store*” as recited in claim 5. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Alternatively, a mail *server* 104’ in Wright can be considered a “*first memory store*” which also can be connected with the FL clients via the internet. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.



Wright Fig. 2.

Wright also discloses that FL clients “need to be able to store and access information.” Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL engine also incorporates a full local database for manipulation and collection of data. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The client can have items stored such as email, a data, or work orders. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Such data or information discloses the “first *record entries* in a file” limitation as recited in claim 5. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Wright at Col. 6, lines 10-21; Fig. 2. As discussed above, Wright discloses a FL client/server system. Wright at Col. 6, lines 10-

21; Fig. 2. Clients, such as a PDA or laptop are connected to the FL *server*. Wright at Col. 6, lines 10-21; Fig. 2. One client, or a mail *server*, connected with the *server* can be considered a “*first memory store*” while another client on the network can be considered a “*second memory store*.” Wright at Col. 6, lines 10-21; Fig. 2. As they are connected by Local area network, or by the internet, they are then “coupled” as recited in claim 5. Wright at Col. 6, lines 10-21; Fig. 2.

Additionally, Wright discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Wright at Col. 6, lines 10-21; Fig. 2. Wright discloses that an application programming interface allows developers to easily extend the FL *server*...to their data sources. Wright at Col. 6, lines 10-21; Fig. 2. Additionally, data can be transferred between the data sources and the PDA's. As a result of such a “session” there is “*synchronizing*” between the data source and the clients. Wright at Col. 6, lines 10-21; Fig. 2. Thus, Wright discloses a “second folder being *synchronized* with a first folder at a first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2. Additionally, after such synchronization, or before any changes are made to the data source or client database, the contents of the data source and FL client would be the same thus disclosing the limitation in claim 5 “a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Wright discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. In particular, and as discussed above, Wright discloses a client/*server* system. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* provides the ability to link hardware devices running the FL engine to access existing enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server*, acts as a “gateway” between FL clients and enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* is a multi-tier client/*server* model in that it creates an intermediate *server* between a client and a traditional, or original *server*. Wright at Col. 6, lines 22-23; Fig. 2. Thus the FL *server* and data sources of Wright can disclose a *global server* as

recited in claim 5. Wright at Col. 6, lines 22-23; Fig. 2. Additionally, as it is a “gateway” and *server* it will transmit data or information between clients and data sources. Wright at Col. 6, lines 22-23; Fig. 2. Therefore, this discloses a “*third memory store* having a third folder for storing third *record entries* in a file.” Wright at Col. 6, lines 22-23; Fig. 2.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Wright discloses an interface for receiving changes at the *first memory store*. Wright at Col. 5, lines 46-52. Wright discloses that because FL clients do not maintain persistent connections with the FL *server*, they must be able to store and access information while not connected to the host database or other data source. Wright at Col. 5, lines 46-52. Therefore, the FL engine included in the FL *server* incorporates a full local database implementation that allows data to be manipulated and collected by the FL client while not connected to the FL *server*. Wright at Col. 5, lines 46-52. Since the data on the client can be manipulated and collected, this discloses the above mentioned limitation. Wright at Col. 5, lines 46-52. Furthermore, as can be seen in Figure 2 of Wright, the display and input buttons on the PDA’s and laptops can be considered an “interface” as recited in claim 5.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Wright discloses a general synchronization module for electronically *synchronizing* the first folder and the third folder such that a new third *record entry* is created in the third folder based on the new first *record entries*. Wright at Col. 5, lines 63-66; Fig. 2. Specifically Wright discloses that the FL *server* can connect with the FL client database, and retrieve data that has been collected by the client since the last connection. Wright at Col. 5, lines 63-66; Fig. 2. Furthermore, as the FL client can be considered to disclose a “first folder” as recited in claim 5, and the FL *server* can be considered to disclose a “third folder” as recited in claim 5, Wright therefore discloses “electronically *synchronizing* the first folder and the third folder.” Wright at Col. 5, lines 63-66; Fig. 2. Alternatively, the *server* can retrieve emails and other data from the email *server*, which can be considered a first memory having a first folder, as discussed above. Wright at Col. 5, lines 63-66; Fig. 2. Additionally because

the data is retrieved from the FL client to the FL *server*, this also discloses “a new third entry is created in the third folder based on the new first *record entries*.” Wright at Col. 5, lines 63-66; Fig. 2. Furthermore the “synchronization module” can be the FL *server*, FL client or the mail *server*. Wright at Col. 5, lines 63-66; Fig. 2.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Wright also discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder such that a new second *record entry* is created in the second folder based on the new third *record entry*. Wright at Col. 5, lines 52-59. In particular, Wright discloses that the FL *server* can query the database on the client in order to add data to the client database, or remove data from the client database in order to make updates to both the client and *server* database to reflect changes that have happened on both sides since the last connection. Wright at Col. 5, lines 52-59. Wright states as a result “[t]hus, a synchronization of the two databases is performed.” Wright at Col. 5, lines 52-59. Therefore as the “third folder” as recited in claim 5 is included on the *server* side, and the “second folder” is included in one of the clients in the network different from the first, this therefore discloses “*synchronizing* the third folder and the second folder.” Wright at Col. 5, lines 52-59. Furthermore as Wright discloses that the *server* is querying the client in order to add data to the client database the new data on the client is created “based on the new third *record entry*”, wherein the third record were updates from the first client as discussed above. Wright at Col. 5, lines 52-59. Furthermore the “synchronization agent” can be the FL *server*, FL client or the mail *server*. Wright at Col. 5, lines 52-59.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder

at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

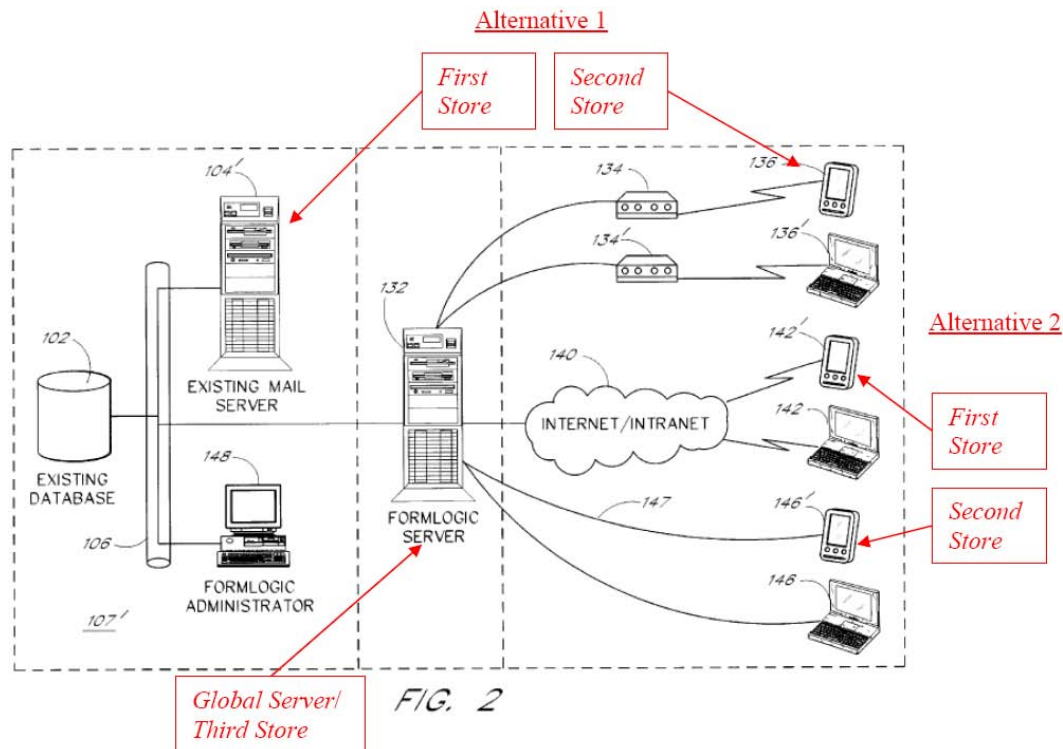
means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Wright.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Wright discloses providing a first memory means including a first folder for storing first *record entries* in a file. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. In particular, Wright discloses a FormLogic (FL) client/*server* system. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL client includes an FL engine which allows FL applications to execute on a variety of handheld devices, such as personal digital assistants (PDA) as well as other portable devices such as laptops. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Therefore the FL client of Wright discloses a “first memory means” as recited in claim 6. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Alternatively, a mail *server* 104’ in Wright can be considered a “first memory means” which also can be connected with the FL clients via the internet. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.



Wright Fig. 2.

Wright also discloses that FL clients “need to be able to store and access information.” Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The FL engine also incorporates a full local database for manipulation and collection of data. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. The client can have items stored such as email, a data, or work orders. Wright at Col. 5, lines 16-24, 46-52; Fig. 2. Such data or information discloses the “first *record entries* in a file” limitation as recited in claim 6. Wright at Col. 5, lines 16-24, 46-52; Fig. 2.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright discloses a second memory means coupled to the *first memory store* the second memory means including a second folder for storing second *record entries* in a file. Wright at Col. 6, lines 10-21; Fig. 2. As discussed above, Wright discloses a FL client/server system. Wright at Col. 6, lines 10-

21; Fig. 2. Clients, such as a PDA or laptop are connected to the FL *server*. Wright at Col. 6, lines 10-21; Fig. 2. One client, or a mail *server*, connected with the *server* can be considered a “first memory means” while another client on the network can be considered a “second memory means.” Wright at Col. 6, lines 10-21; Fig. 2. As they are connected by Local area network, or by the internet, they are then “coupled” as recited in claim 6. Wright at Col. 6, lines 10-21; Fig. 2.

Additionally, Wright discloses the second folder being *synchronized* with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time. Wright at Col. 6, lines 10-21; Fig. 2. Wright discloses that an application programming interface allows developers to easily extend the FL *server*...to their data sources. Wright at Col. 6, lines 10-21; Fig. 2. Additionally, data can be transferred between the data sources and the PDA's. As a result of such a “session” there is “*synchronizing*” between the data source and the clients. Wright at Col. 6, lines 10-21; Fig. 2. Thus, Wright discloses a “second folder being *synchronized* with a first folder at a first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2. Additionally, after such synchronization, or before any changes are made to the data source or client database, the contents of the data source and FL client would be the same thus disclosing the limitation in claim 6 “a content of the second folder is consistent with the content of the first folder at the first moment in time.” Wright at Col. 6, lines 10-21; Fig. 2.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Wright discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. In particular, and as discussed above, Wright discloses a client/*server* system. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* provides the ability to link hardware devices running the FL engine to access existing enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server*, acts as a “gateway” between FL clients and enterprise data sources. Wright at Col. 6, lines 22-23; Fig. 2. The FL *server* is a multi-tier client/*server* model in that it creates an intermediate *server* between a client and a traditional, or original *server*. Wright at Col. 6, lines 22-23; Fig. 2. Thus the FL *server* and data sources of Wright can disclose a *global server* as

recited in claim 6. Wright at Col. 6, lines 22-23; Fig. 2. Additionally, as it is a “gateway” and *server* it will transmit data or information between clients and data sources. Wright at Col. 6, lines 22-23; Fig. 2. Therefore, this discloses a “*third memory store* having a third folder for storing third *record entries* in a file.” Wright at Col. 6, lines 22-23; Fig. 2.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Wright discloses a means for receiving changes at a first memory means. Wright at Col. 5, lines 46-52. Wright discloses that because FL clients do not maintain persistent connections with the FL *server*, they must be able to store and access information while not connected to the host database or other data source. Wright at Col. 5, lines 46-52. Therefore, the FL engine included in the FL *server* incorporates a full local database implementation that allows data to be manipulated and collected by the FL client while not connected to the FL *server*. Wright at Col. 5, lines 46-52. Since the data on the client can be manipulated and collected, this discloses the above mentioned limitation. Wright at Col. 5, lines 46-52.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Wright discloses a means for electronically *synchronizing* the first folder and the third folder such that the content of the changes creates a third *record entry*. Wright at Col. 5, lines 63-66; Fig. 2. Specifically Wright discloses that the FL *server* can connect with the FL client database, and retrieve data that has been collected by the client since the last connection. Wright at Col. 5, lines 63-66; Fig. 2. Furthermore, as the FL client can be considered to disclose a “first folder” as recited in claim 6, and the FL *server* can be considered to disclose a “third folder” as recited in claim 6, Wright therefore discloses “electronically *synchronizing* the first folder and the third folder.” Wright at Col. 5, lines 63-66; Fig. 2. Alternatively, the *server* can retrieve emails and other data from the email *server*, which can be considered a first memory having a first folder, as discussed above. Wright at Col. 5, lines 63-66; Fig. 2.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second

folder based on the new third record entry.” Wright also discloses a means for electronically *synchronizing* the third folder and the second folder such that a new second *record entry* is created in the second folder based on the new third *record entry*. Wright at Col. 5, lines 52-59. In particular, Wright discloses that the FL *server* can query the database on the client in order to add data to the client database, or remove data from the client database in order to make updates to both the client and *server* database to reflect changes that have happened on both sides since the last connection. Wright at Col. 5, lines 52-59. Wright states as a result “[t]hus, a synchronization of the two databases is performed.” Wright at Col. 5, lines 52-59. Therefore as the “third folder” as recited in claim 6 is included on the *server* side, and the “second folder” is included in one of the clients in the network different from the first, this therefore discloses “*synchronizing* the third folder and the second folder.” Wright at Col. 5, lines 52-59. Furthermore as Wright discloses that the *server* is querying the client in order to add data to the client database the new data on the client is created “based on the new third *record entry*”, wherein the third record were updates from the first client as discussed above. Wright at Col. 5, lines 52-59.

I. Claims 1-6 are Anticipated by GloMop Under 35 U.S.C. § 102

**Please see attached Exhibit CC-H
presenting claim charts for
comparison of the GloMop with
claims 1-6 of the ‘231 patent.**

Requester respectfully submits that claims 1-6 of the ‘231 patent are anticipated by GloMop under 35 U.S.C. § 102(b). A claim chart applying GloMop to these claims is submitted herewith as Exhibit CC-H.

1. Claim 1

Claim 1 of the ‘231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in GloMop.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” GloMop discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that the *first memory store* can be a client computer such as an Intel x86 portable running Microsoft Windows or a Macintosh PowerBook running MAC OS. GloMop at p. 5, § 1.6. Therefore, the client computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 1.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” GloMop also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that there can be multiple clients in the GloMop system

as the GloMop system is designed to work with multiple client platforms. GloMop at p. 5, § 1.6. Both the first and second clients can *synchronize* files with the proxy at a first moment in time and will then contain identical first and second folders. GloMop at p. 7, § 2.1.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” GloMop discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. GloMop at p. 2, § 1.2. The proxy *server* is able to store the third folder to data directly connected to the proxy *server* or data remotely connected to the proxy *server*. GloMop at p. 2, § 1.2.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” GloMop discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. GloMop at p. 10, § 3.5. For instance, a new email may be prepared on the first client after the previous synchronization. GloMop at p. 10, § 3.5.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” GloMop discloses electronically *synchronizing* the first folder and the third folder. GloMop at p. 10, § 3.5. As an example, a new email or log file prepared on the first client may be uploaded to the proxy *server*. GloMop at p. 10, § 3.5.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” GloMop discloses electronically *synchronizing* the third folder and the second folder. GloMop at p. 7, § 2.1. The second client would download from the proxy *server* the new email uploaded by the first client to the proxy *server*. GloMop at p. 7, § 2.1.

2. Claim 2

Claim 2 of the '231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in GloMop.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” GloMop discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that the *first memory store* can be a client computer such as an Intel x86 portable running Microsoft Windows or a Macintosh PowerBook running MAC OS. GloMop at p. 5, § 1.6. Therefore, the client computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 2.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” GloMop also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second

folder for storing second *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that there can be multiple clients in the GloMop system as the GloMop system is designed to work with multiple client platforms. GloMop at p. 5, § 1.6. Both the first and second clients can *synchronize* files with the proxy at a first moment in time and will then contain identical first and second folders. GloMop at p. 7, § 2.1.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” GloMop discloses an interface for receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. GloMop at p. 10, § 3.5. For instance, a new email may be prepared on the first client after the previous synchronization. GloMop at p. 10, § 3.5.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” GloMop discloses electronically *synchronizing* the first folder and the third folder. GloMop at p. 10, § 3.5. As an example, a new email or log file prepared on the first client may be uploaded to the proxy *server*. GloMop at p. 10, § 3.5.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” GloMop discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. GloMop at p. 7, § 2.1. The second client would download from the proxy *server* the new email uploaded by the first client to the proxy *server*. GloMop at p. 7, § 2.1.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder

at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in GloMop.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” GloMop discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that the *first memory store* can be a client computer such as an Intel x86 portable running Microsoft Windows or a Macintosh PowerBook running MAC OS. GloMop at p. 5, § 1.6. Therefore, the client computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 3.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” GloMop also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that there can be multiple clients in the GloMop system as the GloMop system is designed to work with multiple client platforms. GloMop at p. 5, § 1.6. Both the first and second clients can *synchronize* files with the proxy at a first moment in time and will then contain identical first and second folders. GloMop at p. 7, § 2.1.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” GloMop discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. GloMop at p. 2, § 1.2. The proxy *server* is able to store the third folder to data directly connected to the proxy *server* or data remotely connected to the proxy *server*. GloMop at p. 2, § 1.2.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” GloMop discloses a means for receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. GloMop at p. 10, § 3.5. For instance, a new email may be prepared on the first client after the previous synchronization. GloMop at p. 10, § 3.5.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” GloMop discloses a means for electronically *synchronizing* the first folder and the third folder. GloMop at p. 10, § 3.5. As an example, a new email or log file prepared on the first client may be uploaded to the proxy *server*. GloMop at p. 10, § 3.5.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” GloMop discloses a means for electronically *synchronizing* the third folder and the second folder. GloMop at p. 7, § 2.1. The second client would download from the proxy *server* the new email uploaded by the first client to the proxy *server*. GloMop at p. 7, § 2.1.

4. Claim 4

Claim 4 of the ‘231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with

the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in GloMop.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” GloMop discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that the *first memory store* can be a client computer such as an Intel x86 portable running Microsoft Windows or a Macintosh PowerBook running MAC OS. GloMop at p. 5, § 1.6. Therefore, the client computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 4.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” GloMop also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that there can be multiple clients in the GloMop system as the GloMop system is designed to work with multiple client platforms. GloMop at p. 5, § 1.6. Both the first and second clients can *synchronize* files with the proxy at a first

moment in time and will then contain identical first and second folders. GloMop at p. 7, § 2.1.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” GloMop discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. GloMop at p. 2, § 1.2. The proxy *server* is able to store the third folder to data directly connected to the proxy *server* or data remotely connected to the proxy *server*. GloMop at p. 2, § 1.2.

The fourth element of claim 4 is “receiving changes at the first memory store.” GloMop discloses receiving changes at the *first memory store*. GloMop at p. 10, § 3.5. For instance, a new email may be prepared on the first client after the previous synchronization. GloMop at p. 10, § 3.5.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” GloMop discloses electronically *synchronizing* the first folder and the third folder. GloMop at p. 10, § 3.5. As an example, a new email or log file prepared on the first client may be uploaded to the proxy *server*. GloMop at p. 10, § 3.5.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” GloMop discloses electronically *synchronizing* the third folder and the second folder. GloMop at p. 7, § 2.1. The second client would download from the proxy *server* the new email uploaded by the first client to the proxy *server*. GloMop at p. 7, § 2.1

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record

entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in GloMop.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” GloMop discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that the *first memory store* can be a client computer such as an Intel x86 portable running Microsoft Windows or a Macintosh PowerBook running MAC OS. GloMop at p. 5, § 1.6. Therefore, the client computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 5.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” GloMop also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that there can be multiple clients in the GloMop system as the GloMop system is designed to work with multiple client platforms. GloMop at p. 5, § 1.6. Both

the first and second clients can *synchronize* files with the proxy at a first moment in time and will then contain identical first and second folders. GloMop at p. 7, § 2.1.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” GloMop discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. GloMop at p. 2, § 1.2. The proxy *server* is able to store the third folder to data directly connected to the proxy *server* or data remotely connected to the proxy *server*. GloMop at p. 2, § 1.2.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” GloMop discloses receiving changes at the first store. GloMop at p. 10, § 3.5. For instance, a new email may be prepared on the first client after the previous synchronization. GloMop at p. 10, § 3.5.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” GloMop discloses electronically *synchronizing* the first folder and the third folder. GloMop at p. 10, § 3.5. As an example, a new email or log file prepared on the first client may be uploaded to the proxy *server*. GloMop at p. 10, § 3.5.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” GloMop discloses electronically *synchronizing* the third folder and the second folder. GloMop at p. 7, § 2.1. The second client would download from the proxy *server* the new email uploaded by the first client to the proxy *server*. GloMop at p. 7, § 2.1.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in GloMop.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” GloMop discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that the *first memory store* can be a client computer such as an Intel x86 portable running Microsoft Windows or a Macintosh PowerBook running MAC OS. GloMop at p. 5, § 1.6. Therefore, the client computer can be a *first memory store*, and the folders and/or documents contained thereon can be *record entries* in a file according to claim 6.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” GloMop also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. GloMop at p. 5, § 1.6. Specifically, GloMop discloses that there can be multiple clients in the GloMop system as the GloMop system is designed to work with multiple client platforms. GloMop at p. 5, § 1.6. Both

the first and second clients can *synchronize* files with the proxy at a first moment in time and will then contain identical first and second folders. GloMop at p. 7, § 2.1.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” GloMop discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. GloMop at p. 2, § 1.2. The proxy *server* is able to store the third folder to data directly connected to the proxy *server* or data remotely connected to the proxy *server*. GloMop at p. 2, § 1.2.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” GloMop discloses receiving changes at the first memory means. GloMop at p. 10, § 3.5. For instance, a new email may be prepared on the first client after the previous synchronization. GloMop at p. 10, § 3.5.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” GloMop discloses electronically *synchronizing* the first folder and the third folder. GloMop at p. 10, § 3.5. As an example, a new email or log file prepared on the first client may be uploaded to the proxy *server*. GloMop at p. 10, § 3.5.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” GloMop discloses electronically *synchronizing* the third folder and the second folder. GloMop at p. 7, § 2.1. The second client would download from the proxy *server* the new email uploaded by the first client to the proxy *server*. GloMop at p. 7, § 2.1.

J. Claims 1-6 are Anticipated by Lambert Under 35 U.S.C. § 102

**Please see attached Exhibit CC-I
presenting claim charts for
comparison of the Lambert with
claims 1-6 of the '231 patent.**

Requester respectfully submits that claims 1-6 of the '231 patent are anticipated by Lambert under 35 U.S.C. § 102(b). A claim chart applying Lambert to these claims is submitted herewith as Exhibit CC-I.

1. Claim 1

Claim 1 of the '231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Lambert.

The first element of claim 1 is "providing a first memory store including a first folder for storing first record entries in a file." Lambert discloses providing a *first*

memory store including a first folder for storing first *record entries* in a file. Lambert at pp. 1-2. Specifically, Lambert discloses that the *first memory store* can be a small resource limited client that stores a user's local mail state. Lambert at pp. 1-2.

The second element of claim 1 is "providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time." Lambert also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lambert at pp. 1-2. Lambert discloses an arbitrary number of users including multiple clients for each user, so the *second memory store* can be a second small resource limited client that also stores a user's local mail state. Lambert at pp. 1-2. The first and second clients are *synchronized* at a first point in time to make sure all clients contain the same changes at that time. Lambert at p. 18.

The third element of claim 1 is "providing a global server including a third memory store having a third folder for storing third record entries in a file." Lambert discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lambert at pp. 1-2. The repository is a powerful resource rich *global server* that stores a user's global mail state. Lambert at pp. 1-2.

The fourth element of claim 1 is "receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time." Lambert discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Lambert at p. 18. Each client maintains an "update list" of messages that have been received at the first client since the last synchronization. Lambert at p. 18.

The fifth element of claim 1 is "electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries." Lambert discloses electronically *synchronizing* the first

folder and the third folder. Lambert at p. 18. The first client will *synchronize* the new changes with the repository. Lambert at p. 18.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lambert discloses electronically *synchronizing* the third folder and the second folder. Lambert at p. 18. The repository then *synchronizes* the changes with the second client because it has not yet received the changes. Lambert at p. 18.

2. Claim 2

Claim 2 of the ‘231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Lambert.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Lambert discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lambert at pp. 1-2. Specifically, Lambert discloses that the *first memory store* can be a small resource limited client that stores a user’s local mail state. Lambert at pp. 1-2.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lambert also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lambert at pp. 1-2. Lambert discloses an arbitrary number of users including multiple clients for each user, so the *second memory store* can be a second small resource limited client that also stores a user’s local mail state. Lambert at pp. 1-2. The first and second clients are *synchronized* at a first point in time to make sure all clients contain the same changes at that time. Lambert at p. 18.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Lambert discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Lambert at p. 18. Each client maintains an “update list” of messages that have been received at the first client since the last synchronization. Lambert at p. 18.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Lambert discloses electronically *synchronizing* the first folder and the third folder. Lambert at p. 18. The first client will *synchronize* the new changes with the repository. Lambert at p. 18.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Lambert discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Lambert at p. 18. The repository then *synchronizes* the changes with the second client because it has not yet received the changes. Lambert at p. 18.

3. Claim 3

Claim 3 of the '231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Lambert.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Lambert discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lambert at pp. 1-2. Specifically, Lambert discloses that the *first memory store* can be a small resource limited client that stores a user’s local mail state. Lambert at pp. 1-2.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lambert also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lambert at pp. 1-2. Lambert discloses an arbitrary number of users including multiple clients for each user, so the *second*

memory store can be a second small resource limited client that also stores a user's local mail state. Lambert at pp. 1-2. The first and second clients are *synchronized* at a first point in time to make sure all clients contain the same changes at that time. Lambert at p. 18.

The third element of claim 3 is "providing a global server including a third memory means having a third folder for storing third record entries in a file." Lambert discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lambert at pp. 1-2. The repository is powerful resource rich *global server* that stores a user's global mail state. Lambert at pp. 1-2.

The fourth element of claim 3 is "means for receiving new record file entries at the first memory means." Lambert discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Lambert at p. 18. Each client maintains an "update list" of messages that have been received at the first client since the last synchronization. Lambert at p. 18.

The fifth element of claim 3 is "means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries." Lambert discloses electronically *synchronizing* the first folder and the third folder. Lambert at p. 18. The first client will *synchronize* the new changes with the repository. Lambert at p. 18.

The last element of claim 3 is "means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry." Lambert discloses electronically *synchronizing* the third folder and the second folder. Lambert at p. 18. The repository then *synchronizes* the changes with the second client because it has not yet received the changes. Lambert at p. 18.

4. Claim 4

Claim 4 of the '231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Lambert.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Lambert discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lambert at pp. 1-2. Specifically, Lambert discloses that the *first memory store* can be a small resource limited client that stores a user’s local mail state. Lambert at pp. 1-2.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lambert also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lambert at pp. 1-2. Lambert discloses an arbitrary number of users including multiple clients for each user, so the *second memory store* can be a second small resource limited client that also stores a user’s local mail state. Lambert at pp. 1-2. The first and second clients are *synchronized* at a first point in time to make sure all clients contain the same changes at that time. Lambert at p. 18.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lambert discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lambert at pp. 1-2. The repository is powerful resource rich *global server* that stores a user’s global mail state. Lambert at pp. 1-2.

The fourth element of claim 4 is “receiving changes at the first memory store.” Lambert discloses receiving changes at the *first memory store*. Lambert at p. 18. Each client maintains an “update list” of messages that have been received at the first client since the last synchronization. Lambert at p. 18.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lambert discloses electronically *synchronizing* the first folder and the third folder. Lambert at p. 18. The first client will *synchronize* the new changes with the repository. Lambert at p. 18.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lambert discloses electronically *synchronizing* the third folder and the second folder. Lambert at p. 18. The repository then *synchronizes* the changes with the second client because it has not yet received the changes. Lambert at p. 18.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Lambert.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Lambert discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lambert at pp. 1-2. Specifically, Lambert discloses that the *first memory store* can be a small resource limited client that stores a user’s local mail state. Lambert at pp. 1-2.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lambert also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lambert at pp. 1-2. Lambert discloses an arbitrary number of users including multiple clients for each user, so the *second memory store* can be a second small resource limited client that also stores a user’s local mail state. Lambert at pp. 1-2. The first and second clients are *synchronized* at a first point in time to make sure all clients contain the same changes at that time. Lambert at p. 18.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Lambert discloses providing a *global server* including a *third memory store* having a third folder

for storing third *record entries* in a file. Lambert at pp. 1-2. The repository is powerful resource rich *global server* that stores a user's global mail state. Lambert at pp. 1-2.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Lambert discloses receiving changes at the *first memory store*. Lambert at p. 18. Each client maintains an “update list” of messages that have been received at the first client since the last synchronization. Lambert at p. 18.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lambert discloses electronically *synchronizing* the first folder and the third folder. Lambert at p. 18. The first client will *synchronize* the new changes with the repository. Lambert at p. 18.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Lambert discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Lambert at p. 18. The repository then *synchronizes* the changes with the second client because it has not yet received the changes. Lambert at p. 18.

6. Claim 6

Claim 6 of the '231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Lambert.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Lambert discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Lambert at pp. 1-2. Specifically, Lambert discloses that the *first memory store* can be a small resource limited client that stores a user’s local mail state. Lambert at pp. 1-2.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Lambert also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Lambert at pp. 1-2. Lambert discloses an arbitrary number of users including multiple clients for each user, so the *second memory store* can be a second small resource limited client that also stores a user’s local mail state. Lambert at pp. 1-2. The first and second clients are *synchronized* at a first point in time to make sure all clients contain the same changes at that time. Lambert at p. 18.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Lambert discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Lambert at pp. 1-2. The repository is powerful resource rich *global server* that stores a user’s global mail state. Lambert at pp. 1-2.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Lambert discloses receiving changes at the *first memory store*. Lambert at p.

18. Each client maintains an “update list” of messages that have been received at the first client since the last synchronization. Lambert at p. 18.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Lambert discloses electronically *synchronizing* the first folder and the third folder. Lambert at p. 18. The first client will *synchronize* the new changes with the repository. Lambert at p. 18.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Lambert discloses electronically *synchronizing* the third folder and the second folder. Lambert at p. 18. The repository then *synchronizes* the changes with the second client because it has not yet received the changes. Lambert at p. 18.

**K. Claims 1-6 are Anticipated by Satyanarayanan Under
35 U.S.C. § 102**

**Please see attached Exhibit CC-J
presenting claim charts for
comparison of the Satyanarayanan
with claims 1-6 of the ‘231 patent.**

Requester respectfully submits that claims 1-6 of the ‘231 patent are anticipated by Satyanarayanan under 35 U.S.C. § 102(b). A claim chart applying Satyanarayanan to these claims is submitted herewith as Exhibit CC-J.

1. Claim 1

Claim 1 of the ‘231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with

the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Satyanarayanan.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Satyanarayanan discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Satyanarayanan at p. 30. Specifically, Satyanarayanan discloses that a client, such as an IBM ThinkPad, contains a cache manager, Venus, to store data. Satyanarayanan at p. 30. The volumes are files and the volume objects are *record entries*. Satyanarayanan at 28.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Satyanarayanan also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Satyanarayanan at pp. 28 and 30. Satyanarayanan discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Satyanarayanan at pp. 28 and 30. Both the first and second clients *synchronize* with each other through the *server* by hoarding data while connected so that they contain the same data. Satyanarayanan at p. 28.

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Satyanarayanan discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Satyanarayanan at pp. 27-28. Satyanarayanan discloses file *servers* mapped to a single namespace so that clients view them as a single location. Satyanarayanan at pp. 27-28.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Satyanarayanan discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Satyanarayanan at p. 29. While disconnected, the first client logs changes to the data stored on it with volume stamps. Satyanarayanan at p. 29.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Satyanarayanan discloses electronically *synchronizing* the first folder and the third folder. Satyanarayanan at p. 28. When the first client reconnects to the Coda system, it *resynchronizes* its changed data with the file *server*. Satyanarayanan at p. 28.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Satyanarayanan discloses electronically *synchronizing* the third folder and the second folder. Satyanarayanan at p. 28. The second client *synchronizes* with the file *server* and integrates any new changes on the file *server* with the data on the second client. Satyanarayanan at p. 28.

2. Claim 2

Claim 2 of the ‘231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Satyanarayanan.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Satyanarayanan discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Satyanarayanan at p. 30. Specifically, Satyanarayanan discloses that a client, such as an IBM ThinkPad, contains a cache manager, Venus, to store data. Satyanarayanan at p. 30. The volumes are files and the volume objects are *record entries*. Satyanarayanan at 28.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Satyanarayanan also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Satyanarayanan at pp. 28 and 30. Satyanarayanan discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Satyanarayanan at pp. 28 and 30. Both the first and second clients *synchronize* with each other through the *server* by hoarding data while connected so that they contain the same data. Satyanarayanan at p. 28.

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Satyanarayanan discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Satyanarayanan at p. 29. While disconnected, the first client logs changes to the data stored on it with volume stamps. Satyanarayanan at p. 29.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Satyanarayanan discloses electronically *synchronizing* the first folder and the third folder. Satyanarayanan at p. 28. When the first client reconnects to the Coda system, it *resynchronizes* its changed data with the file *server*. Satyanarayanan at p. 28.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Satyanarayanan discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Satyanarayanan at p. 28. The second client *synchronizes* with the file *server* and integrates any new changes on the file *server* with the data on the second client. Satyanarayanan at p. 28.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Satyanarayanan.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Satyanarayanan discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Satyanarayanan at p. 30. Specifically, Satyanarayanan discloses that a client, such as an IBM ThinkPad, contains a cache manager, Venus, to store data. Satyanarayanan at p. 30. The volumes are files and the volume objects are *record entries*. Satyanarayanan at 28.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Satyanarayanan also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Satyanarayanan at pp. 28 and 30. Satyanarayanan discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Satyanarayanan at pp. 28 and 30. Both the first and second clients *synchronize* with each other through the *server* by hoarding data while connected so that they contain the same data. Satyanarayanan at p. 28.

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Satyanarayanan discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Satyanarayanan at pp. 27-

28. Satyanarayanan discloses file *servers* mapped to a single namespace so that clients view them as a single location. Satyanarayanan at pp. 27-28.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Satyanarayanan discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Satyanarayanan at p. 29. While disconnected, the first client logs changes to the data stored on it with volume stamps. Satyanarayanan at p. 29.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Satyanarayanan discloses electronically *synchronizing* the first folder and the third folder. Satyanarayanan at p. 28. When the first client reconnects to the Coda system, it *resynchronizes* its changed data with the file *server*. Satyanarayanan at p. 28.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Satyanarayanan discloses electronically *synchronizing* the third folder and the second folder. Satyanarayanan at p. 28. The second client *synchronizes* with the file *server* and integrates any new changes on the file *server* with the data on the second client. Satyanarayanan at p. 28.

4. Claim 4

Claim 4 of the ‘231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Satyanarayanan.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Satyanarayanan discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Satyanarayanan at p. 30. Specifically, Satyanarayanan discloses that a client, such as an IBM ThinkPad, contains a cache manager, Venus, to store data. Satyanarayanan at p. 30. The volumes are files and the volume objects are *record entries*. Satyanarayanan at 28.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Satyanarayanan also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Satyanarayanan at pp. 28 and 30. Satyanarayanan discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Satyanarayanan at pp. 28 and 30. Both the first and second clients *synchronize* with each other through the *server* by hoarding data while connected so that they contain the same data. Satyanarayanan at p. 28.

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Satyanarayanan discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Satyanarayanan at pp. 27-28. Satyanarayanan discloses file *servers* mapped to a single namespace so that clients view them as a single location. Satyanarayanan at pp. 27-28.

The fourth element of claim 4 is “receiving changes at the first memory store.” Satyanarayanan discloses receiving changes at the *first memory store*. Satyanarayanan at p. 29. While disconnected, the first client logs changes to the data stored on it with volume stamps. Satyanarayanan at p. 29.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Satyanarayanan discloses electronically *synchronizing* the first folder and the third folder. Satyanarayanan at p. 28. When the first client reconnects to the Coda system, it *resynchronizes* its changed data with the file *server*. Satyanarayanan at p. 28.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Satyanarayanan discloses electronically *synchronizing* the third folder and the second folder. Satyanarayanan at p. 28. The second client *synchronizes* with the file *server* and integrates any new changes on the file *server* with the data on the second client. Satyanarayanan at p. 28.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Satyanarayanan.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Satyanarayanan discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Satyanarayanan at p. 30. Specifically, Satyanarayanan discloses that a client, such as an IBM ThinkPad, contains a cache manager, Venus, to store data. Satyanarayanan at p. 30. The volumes are files and the volume objects are *record entries*. Satyanarayanan at 28.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Satyanarayanan also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Satyanarayanan at pp. 28 and 30. Satyanarayanan discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Satyanarayanan at pp. 28 and 30. Both the first and second clients *synchronize* with each other through the *server* by hoarding data while connected so that they contain the same data. Satyanarayanan at p. 28.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Satyanarayanan discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Satyanarayanan at pp. 27-28. Satyanarayanan discloses file *servers* mapped to a single namespace so that clients view them as a single location. Satyanarayanan at pp. 27-28.

The fourth element of claim 5 is “an interface for receiving changes at the first

memory store.” Satyanarayanan discloses receiving changes at the *first memory store*. Satyanarayanan at p. 29. While disconnected, the first client logs changes to the data stored on it with volume stamps. Satyanarayanan at p. 29.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Satyanarayanan discloses electronically *synchronizing* the first folder and the third folder. Satyanarayanan at p. 28. When the first client reconnects to the Coda system, it *resynchronizes* its changed data with the file *server*. Satyanarayanan at p. 28.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Satyanarayanan discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Satyanarayanan at p. 28. The second client *synchronizes* with the file *server* and integrates any new changes on the file *server* with the data on the second client. Satyanarayanan at p. 28.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Satyanarayanan.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Satyanarayanan discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Satyanarayanan at p. 30. Specifically, Satyanarayanan discloses that a client, such as an IBM ThinkPad, contains a cache manager, Venus, to store data. Satyanarayanan at p. 30. The volumes are files and the volume objects are *record entries*. Satyanarayanan at 28.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Satyanarayanan also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Satyanarayanan at pp. 28 and 30. Satyanarayanan discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Satyanarayanan at pp. 28 and 30. Both the first and second clients *synchronize* with each other through the *server* by hoarding data while connected so that they contain the same data. Satyanarayanan at p. 28.

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Satyanarayanan discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Satyanarayanan at pp. 27-28. Satyanarayanan discloses file *servers* mapped to a single namespace so that clients view them as a single location. Satyanarayanan at pp. 27-28.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Satyanarayanan discloses receiving changes at the *first memory store*.

Satyanarayanan at p. 29. While disconnected, the first client logs changes to the data stored on it with volume stamps. Satyanarayanan at p. 29.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Satyanarayanan discloses electronically *synchronizing* the first folder and the third folder. Satyanarayanan at p. 28. When the first client reconnects to the Coda system, it *resynchronizes* its changed data with the file *server*. Satyanarayanan at p. 28.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Satyanarayanan discloses electronically *synchronizing* the third folder and the second folder. Satyanarayanan at p. 28. The second client *synchronizes* with the file *server* and integrates any new changes on the file *server* with the data on the second client. Satyanarayanan at p. 28.

**L. Claims 1-6 are Anticipated by Kumar Under
35 U.S.C. § 102**

**Please see attached Exhibit CC-K
presenting claim charts for
comparison of the Kumar with claims
1-6 of the ‘231 patent.**

Requester respectfully submits that claims 1-6 of the ‘231 patent are anticipated by Kumar under 35 U.S.C. § 102(b). A claim chart applying Kumar to these claims is submitted herewith as Exhibit CC-K.

1. Claim 1

Claim 1 of the ‘231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with

the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

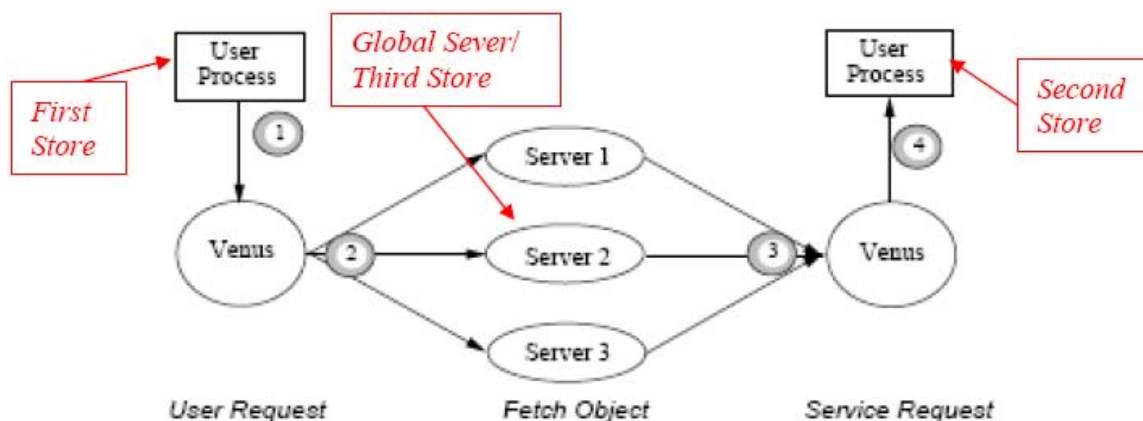
electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Kumar.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Kumar discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kumar at pp. 7-8. Specifically, Kumar discloses the *first memory store* is a client and that data such as directories and files are cached on the client. Kumar at pp. 7-8.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kumar also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kumar at pp. 7-8 and Fig 2.2(a). Kumar discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kumar at 7 and Fig. 2.2(a). Both the first and second clients *synchronize* with each other through the *servers* while connected so that they contain the same data. Kumar at p. 8.



Kumar Fig 2.2(a)

The third element of claim 1 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Kumar discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kumar at pp. 7-8. Kumar discloses a *global server* is one *server* of the set of *servers* that make up Vice. Kumar at pp. 7-8.

The fourth element of claim 1 is “receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time.” Kumar discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Kumar at p. 8. Kumar discloses the file systems at the clients can be later updated, requiring the updated data to be sent to the *servers*. Kumar at p. 8.

The fifth element of claim 1 is “electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries.” Kumar discloses electronically *synchronizing* the first folder and the third folder. Kumar at p. 8. Kumar discloses when data is updated at the first client, it must be sent to and *synchronized* with the *servers*. Kumar at p. 8.

The last element of claim 1 is “electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Kumar discloses electronically *synchronizing* the third folder and the second folder. Kumar at p. 8. Kumar discloses that when the *servers* receive updated data from one client, that the *servers* then inform the other clients storing

that data that the stored data is no longer valid and then those clients with the old data *synchronize* with the *servers* to obtain the new updated data. Kumar at 8.

2. Claim 2

Claim 2 of the '231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

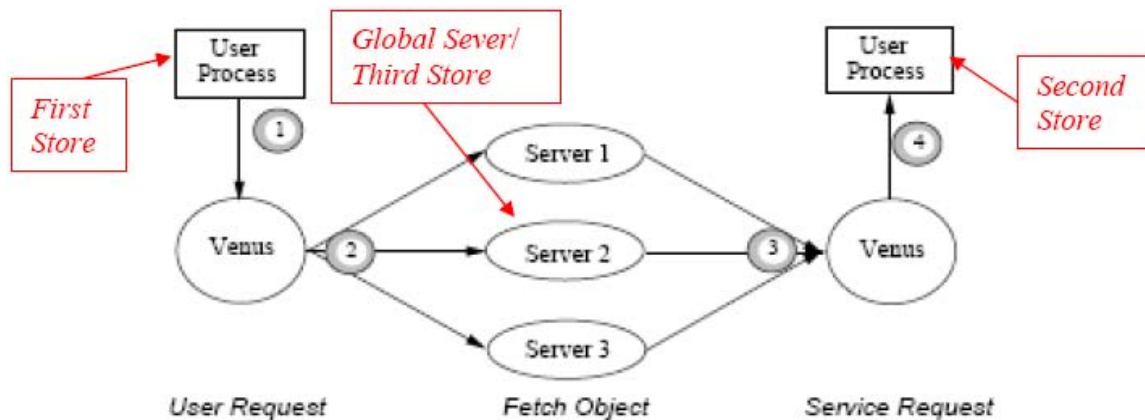
a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Kumar.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Kumar discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kumar at pp. 7-8. Specifically, Kumar discloses the *first memory store* is a client and that data such as directories and files are cached on the client. Kumar at pp. 7-8.

The second element of claim 2 is “a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kumar also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for

storing second *record entries* in a file. Kumar at pp. 7-8 and Fig 2.2(a). Kumar discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kumar at 7 and Fig. 2.2(a). Both the first and second clients *synchronize* with each other through the *servers* while connected so that they contain the same data. Kumar at p. 8.



Kumar Fig 2.2(a)

The third element of claim 2 is “an interface for receiving new record entries in a file at the first memory store.” Kumar discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Kumar at p. 8. Kumar discloses the file systems at the clients can be later updated, requiring the updated data to be sent to the *servers*. Kumar at p. 8.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Kumar discloses electronically *synchronizing* the first folder and the third folder. Kumar at p. 8. Kumar discloses when data is updated at the first client, it must be sent to and *synchronized* with the *servers*. Kumar at p. 8.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Kumar discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Kumar at p. 8. Kumar discloses that when the *servers* receive updated data from

one client, that the *servers* then inform the other clients storing that data that the stored data is no longer valid and then those clients with the old data *synchronize* with the *servers* to obtain the new updated data. Kumar at 8.

3. Claim 3

Claim 3 of the '231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

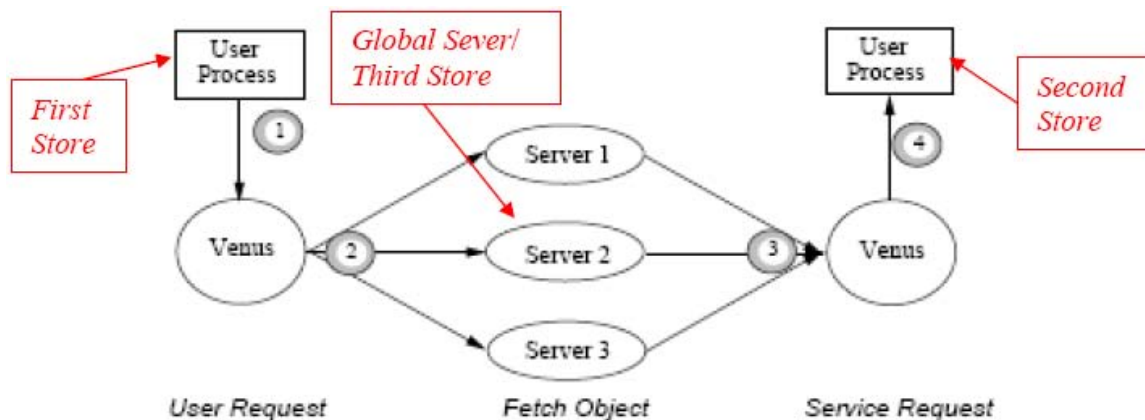
means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Kumar.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Kumar discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kumar at pp. 7-8. Specifically, Kumar discloses the *first memory store* is a client and that data such as directories and files are cached on the client. Kumar at pp. 7-8.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first

moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kumar also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kumar at pp. 7-8 and Fig 2.2(a). Kumar discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kumar at 7 and Fig. 2.2(a). Both the first and second clients *synchronize* with each other through the *servers* while connected so that they contain the same data. Kumar at p. 8.



Kumar Fig 2.2(a)

The third element of claim 3 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Kumar discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kumar at pp. 7-8. Kumar discloses a *global server* is one *server* of the set of *servers* that make up Vice. Kumar at pp. 7-8.

The fourth element of claim 3 is “means for receiving new record file entries at the first memory means.” Kumar discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Kumar at p. 8. Kumar discloses the file systems at the clients can be later updated, requiring the updated data to be sent to the *servers*. Kumar at p. 8.

The fifth element of claim 3 is “means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries.” Kumar discloses electronically *synchronizing* the

first folder and the third folder. Kumar at p. 8. Kumar discloses when data is updated at the first client, it must be sent to and *synchronized* with the *servers*. Kumar at p. 8.

The last element of claim 3 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Kumar discloses electronically *synchronizing* the third folder and the second folder. Kumar at p. 8. Kumar discloses that when the *servers* receive updated data from one client, that the *servers* then inform the other clients storing that data that the stored data is no longer valid and then those clients with the old data *synchronize* with the *servers* to obtain the new updated data. Kumar at 8.

4. Claim 4

Claim 4 of the ‘231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

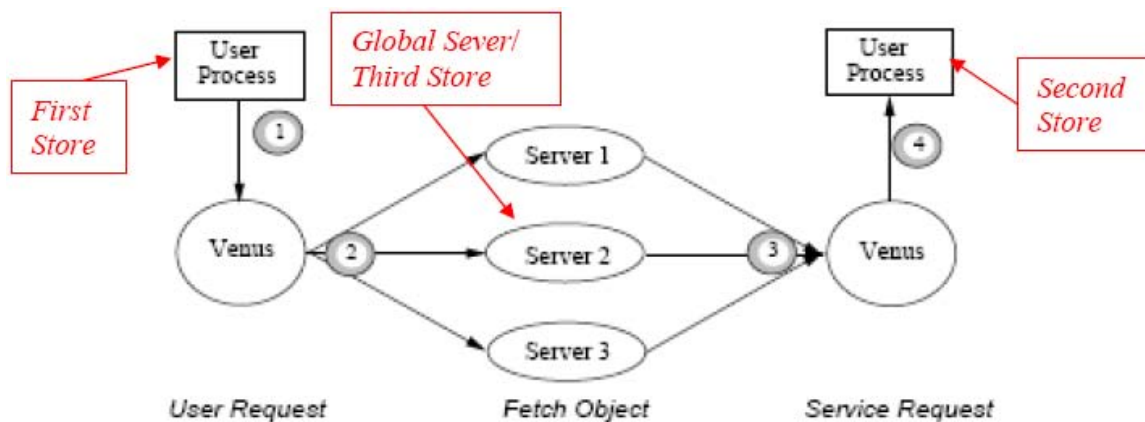
electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Kumar.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Kumar discloses providing a *first memory*

store including a first folder for storing first *record entries* in a file. Kumar at pp. 7-8. Specifically, Kumar discloses the *first memory store* is a client and that data such as directories and files are cached on the client. Kumar at pp. 7-8.

The second element of claim 4 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kumar also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kumar at pp. 7-8 and Fig 2.2(a). Kumar discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kumar at 7 and Fig. 2.2(a). Both the first and second clients *synchronize* with each other through the *servers* while connected so that they contain the same data. Kumar at p. 8.



Kumar Fig 2.2(a)

The third element of claim 4 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Kumar discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kumar at pp. 7-8. Kumar discloses a *global server* is one *server* of the set of *servers* that make up Vice. Kumar at pp. 7-8.

The fourth element of claim 4 is “receiving changes at the first memory store.” Kumar discloses receiving changes at the *first memory store*. Kumar at p. 8. Kumar

discloses the file systems at the clients can be later updated, requiring the updated data to be sent to the *servers*. Kumar at p. 8.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Kumar discloses electronically *synchronizing* the first folder and the third folder. Kumar at p. 8. Kumar discloses when data is updated at the first client, it must be sent to and *synchronized* with the *servers*. Kumar at p. 8.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Kumar discloses electronically *synchronizing* the third folder and the second folder. Kumar at p. 8. Kumar discloses that when the *servers* receive updated data from one client, that the *servers* then inform the other clients storing that data that the stored data is no longer valid and then those clients with the old data *synchronize* with the *servers* to obtain the new updated data. Kumar at 8.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

an interface for receiving changes at the first memory store;

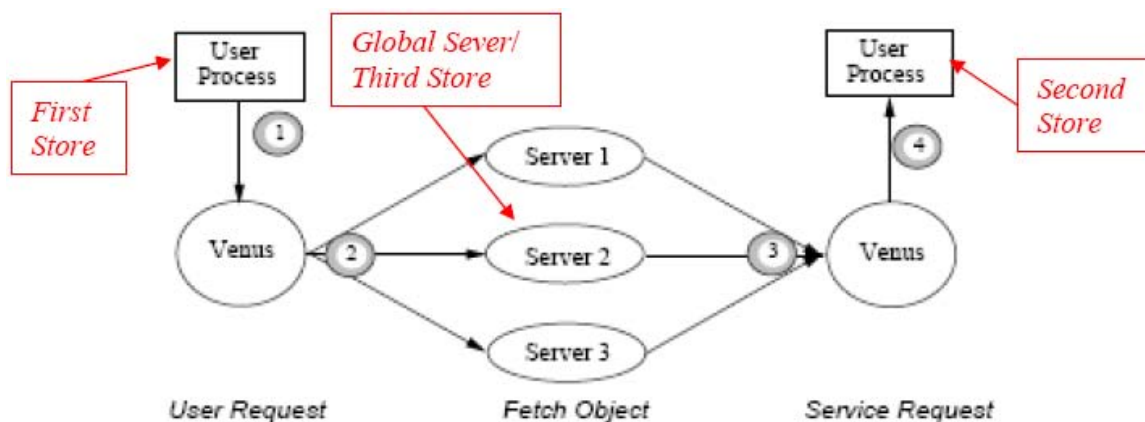
a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Kumar.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Kumar discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kumar at pp. 7-8. Specifically, Kumar discloses the *first memory store* is a client and that data such as directories and files are cached on the client. Kumar at pp. 7-8.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kumar also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kumar at pp. 7-8 and Fig 2.2(a). Kumar discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kumar at 7 and Fig. 2.2(a). Both the first and second clients *synchronize* with each other through the *servers* while connected so that they contain the same data. Kumar at p. 8.



Kumar Fig 2.2(a)

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Kumar discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kumar at pp. 7-8. Kumar discloses a *global server* is one *server* of the set of *servers* that make up Vice. Kumar at pp. 7-8.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Kumar discloses receiving changes at the *first memory store*. Kumar at p. 8. Kumar discloses the file systems at the clients can be later updated, requiring the updated data to be sent to the *servers*. Kumar at p. 8.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Kumar discloses electronically *synchronizing* the first folder and the third folder. Kumar at p. 8. Kumar discloses when data is updated at the first client, it must be sent to and *synchronized* with the *servers*. Kumar at p. 8.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Kumar discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Kumar at p. 8. Kumar discloses that when the *servers* receive updated data from one client, that the *servers* then inform the other clients storing that data that the stored data is no longer valid and then those clients with the old data *synchronize* with the *servers* to obtain the new updated data. Kumar at 8.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder

at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

means for receiving changes at the first memory means;

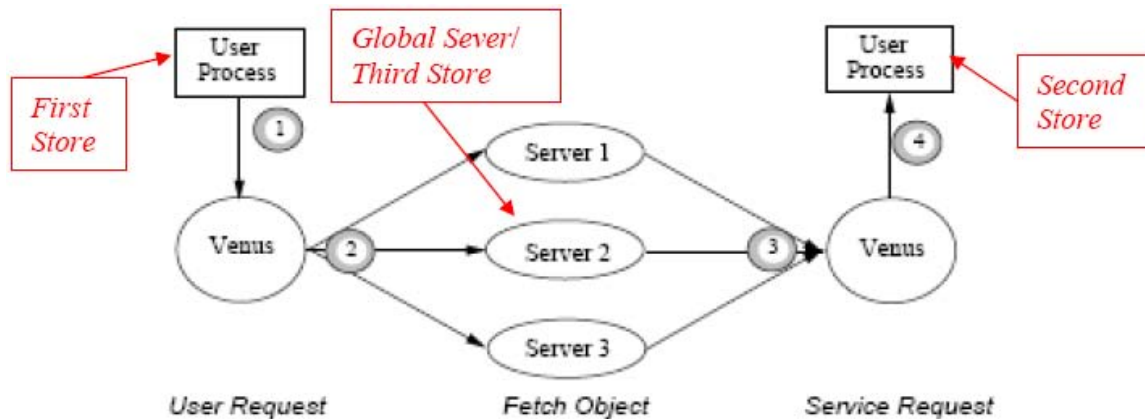
means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Kumar.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Kumar discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kumar at pp. 7-8. Specifically, Kumar discloses the *first memory store* is a client and that data such as directories and files are cached on the client. Kumar at pp. 7-8.

The second element of claim 6 is “second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kumar also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kumar at pp. 7-8 and Fig 2.2(a). Kumar discloses that multiple clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kumar at 7 and Fig. 2.2(a). Both the first and second clients *synchronize* with each other through the *servers* while connected so that they contain the same data. Kumar at p. 8.



Kumar Fig 2.2(a)

The third element of claim 6 is “providing a global server including a third memory means having a third folder for storing third record entries in a file.” Kumar discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kumar at pp. 7-8. Kumar discloses a *global server* is one *server* of the set of *servers* that make up Vice. Kumar at pp. 7-8.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Kumar discloses receiving changes at the *first memory store*. Kumar at p. 8. Kumar discloses the file systems at the clients can be later updated, requiring the updated data to be sent to the *servers*. Kumar at p. 8.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Kumar discloses electronically *synchronizing* the first folder and the third folder. Kumar at p. 8. Kumar discloses when data is updated at the first client, it must be sent to and *synchronized* with the *servers*. Kumar at p. 8.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Kumar discloses electronically *synchronizing* the third folder and the second folder. Kumar at p. 8. Kumar discloses that when the *servers* receive updated data from one client, that the *servers* then inform the other clients storing that data that the stored data is no longer valid and then those

clients with the old data *synchronize* with the *servers* to obtain the new updated data.
Kumar at 8.

**M. Claims 1-6 are Anticipated by Kistler Under
35 U.S.C. § 102**

**Please see attached Exhibit CC-L
presenting claim charts for
comparison of the Kistler with claims
1-6 of the '231 patent.**

Requester respectfully submits that claims 1-6 of the '231 patent are anticipated by Kistler under 35 U.S.C. § 102(b). A claim chart applying Kistler to these claims is submitted herewith as Exhibit CC-L.

1. Claim 1

Claim 1 of the '231 patent reads as follows:

1. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

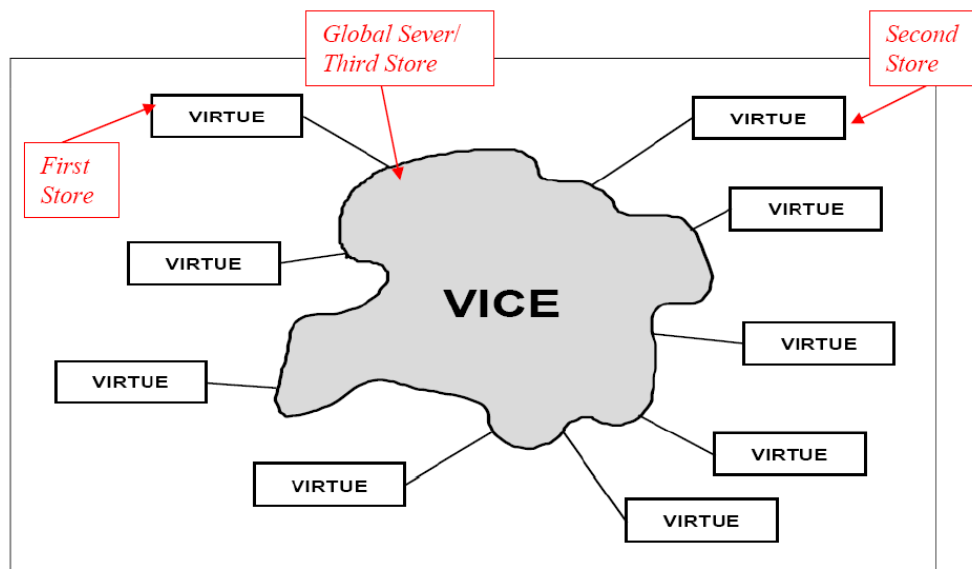
receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time;

electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries; and

electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 1 are disclosed in Kistler.

The first element of claim 1 is “providing a first memory store including a first folder for storing first record entries in a file.” Kistler discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kistler at pp. 14, 16-18 and Fig 2.1. Specifically, Kistler discloses that the first data stores are caches on user workstations called Virtue that maintain folders in the file system. Kistler at pp. 14, 16 and Fig. 2.1. Kistler also discloses storing *record entries* in a file since only an object’s descriptors can be stored instead of the entire object. Kistler at pp 17-18.



The structure labeled "Vice" is a collection of trusted servers and untrusted networks. The nodes labeled "Virtue" are private or public workstations (or occasionally time-sharing systems). Vice is reputed to stand for "Vast Integrated Computing Environment," and Virtue for "Virtue is Realized through Unix and Emacs."

Kistler Fig. 2.1 at pg. 14.

The second element of claim 1 is “providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kistler also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kistler at pp. 14, 16-19 and Fig 2.1. Kistler discloses that multiple Virtue clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kistler at pp.

14, 16 and Fig 2.1. In addition the second client is also able to store a folder with *record entries* in a file since it can store an object's descriptors instead of the entire object. Kistler at 17-18. Finally, the system is able to determine when one client has updated its cache and in order to notify the other clients to *synchronize* by updating their caches as well. Kistler at p. 19.

The third element of claim 1 is "providing a global server including a third memory store having a third folder for storing third record entries in a file." Kistler discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kistler at p. 14. Kistler discloses Vice is a *global server* that is made up of a number of *servers* that appear to the Virtue clients as one single *server*. Kistler at p. 14.

The fourth element of claim 1 is "receiving new first record entries in a file at the first memory store at a second moment in time later than the first moment in time." Kistler discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Kistler at p. 19. Kistler discloses a client may update cached objects at a later time. Kistler at p. 19.

The fifth element of claim 1 is "electronically synchronizing the first folder and the third folder, such that a new third record entry is created in the third folder based on the new first record entries." Kistler discloses electronically *synchronizing* the first folder and the third folder. Kistler at p. 18. Kistler discloses that when an object is updated on a client, the updated object is *synchronized* to the *global server*, Vice. Kistler at p. 18.

The last element of claim 1 is "electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry." Kistler discloses electronically *synchronizing* the third folder and the second folder. Kistler at p. 18-19. Kistler discloses the *global server*, Vice, informs clients when their objects are outdated and then the *global server* and the outdated client *synchronizes* so the client gets the updated objects. Kistler at p. 18-19.

2. Claim 2

Claim 2 of the '231 patent reads as follows:

2. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

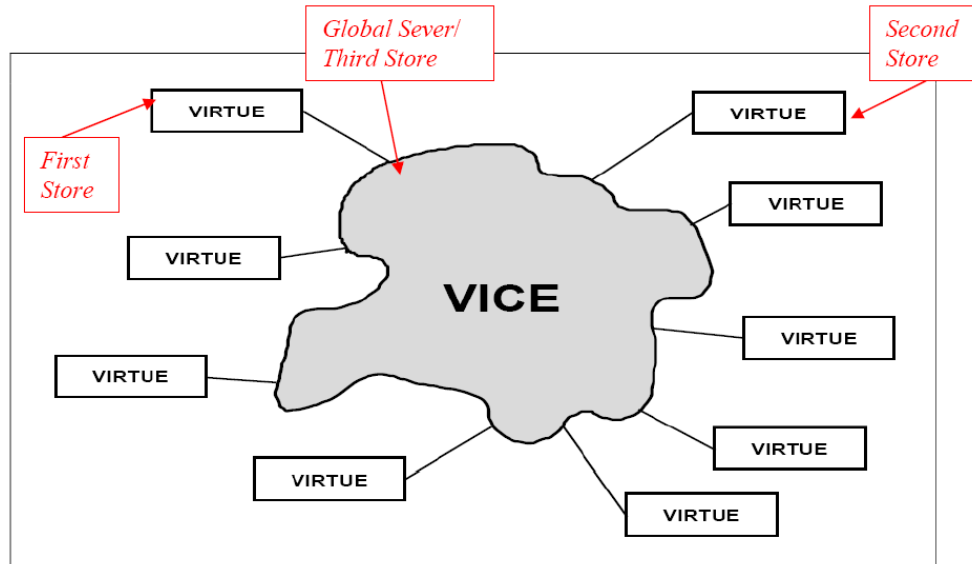
an interface for receiving new record entries in a file at the first memory store;

a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 2 are disclosed in Kistler.

The first element of claim 2 is “a first memory store including a first folder for storing first record entries in a file.” Kistler discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kistler at pp. 14, 16-18 and Fig 2.1. Specifically, Kistler discloses that the first data stores are caches on user workstations called Virtue that maintain folders in the file system. Kistler at pp. 14, 16 and Fig. 2.1. Kistler also discloses storing *record entries* in a file since only an object’s descriptors can be stored instead of the entire object. Kistler at pp 17-18.



The structure labeled "Vice" is a collection of trusted servers and untrusted networks. The nodes labeled "Virtue" are private or public workstations (or occasionally time-sharing systems). Vice is reputed to stand for "Vast Integrated Computing Environment," and Virtue for "Virtue is Realized through Unix and Emacs."

Kistler Fig. 2.1 at pg. 14.

The second element of claim 2 is "a second memory store coupled to the first memory store the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time." Kistler also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kistler at pp. 14, 16-19 and Fig 2.1. Kistler discloses that multiple Virtue clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kistler at pp. 14, 16 and Fig 2.1. In addition the second client is also able to store a folder with *record entries* in a file since it can store an object's descriptors instead of the entire object. Kistler at 17-18. Finally, the system is able to determine when one client has updated its cache and in order to notify the other clients to *synchronize* by updating their caches as well. Kistler at p. 19.

The third element of claim 2 is "an interface for receiving new record entries in a file at the first memory store." Kistler discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in

time. Kistler at p. 19. Kistler discloses a client may update cached objects at a later time. Kistler at p. 19.

The fourth element of claim 2 is “a general synchronization module for electronically synchronizing the first folder and a third folder such that a new third record entry is created in the third folder based on the new first record entries.” Kistler discloses electronically *synchronizing* the first folder and the third folder. Kistler at p. 18. Kistler discloses that when an object is updated on a client, the updated object is synchronized to the global server, Vice. Kistler at p. 18.

The last element of claim 2 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the new third record entry.” Kistler discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Kistler at p. 18-19. Kistler discloses the *global server*, Vice, informs clients when their objects are outdated and then the *global server* and the outdated client *synchronizes* so the client gets the updated objects. Kistler at p. 18-19.

3. Claim 3

Claim 3 of the ‘231 patent reads as follows:

3. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

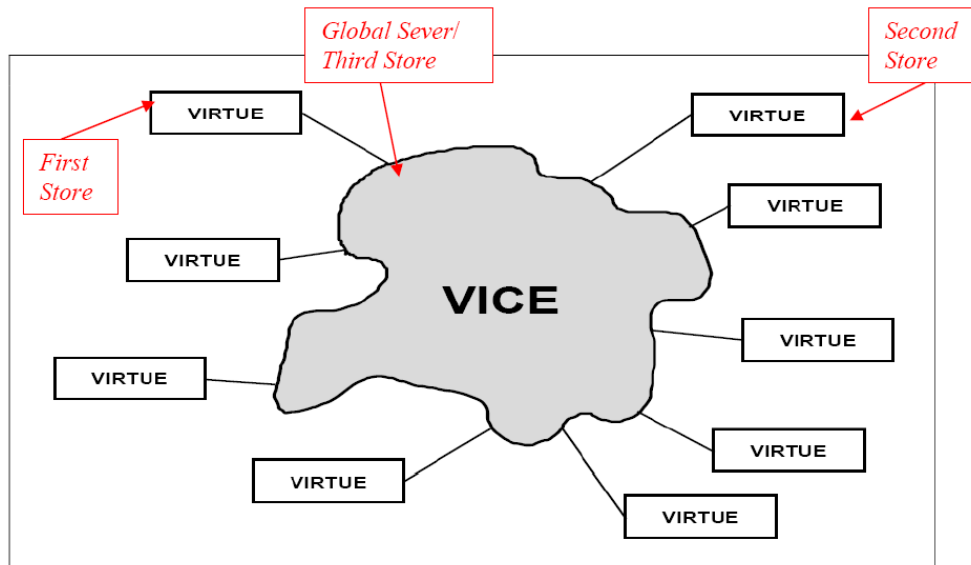
means for receiving new record file entries at the first memory means;

means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 3 are disclosed in Kistler.

The first element of claim 3 is “first memory means including a first folder for storing first record entries in a file.” Kistler discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kistler at pp. 14, 16-18 and Fig 2.1. Specifically, Kistler discloses that the first data stores are caches on user workstations called Virtue that maintain folders in the file system. Kistler at pp. 14, 16 and Fig. 2.1. Kistler also discloses storing *record entries* in a file since only an object’s descriptors can be stored instead of the entire object. Kistler at pp 17-18.



The structure labeled “Vice” is a collection of trusted servers and untrusted networks. The nodes labeled “Virtue” are private or public workstations (or occasionally time-sharing systems). Vice is reputed to stand for “Vast Integrated Computing Environment,” and Virtue for “Virtue is Realized through Unix and Emacs.”

Kistler Fig. 2.1 at pg. 14.

The second element of claim 3 is “second memory means coupled to the first memory means, the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kistler also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for

storing second *record entries* in a file. Kistler at pp. 14, 16-19 and Fig 2.1. Kistler discloses that multiple Virtue clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kistler at pp. 14, 16 and Fig 2.1. In addition the second client is also able to store a folder with *record entries* in a file since it can store an object's descriptors instead of the entire object. Kistler at 17-18. Finally, the system is able to determine when one client has updated its cache and in order to notify the other clients to *synchronize* by updating their caches as well. Kistler at p. 19.

The third element of claim 3 is "providing a global server including a third memory means having a third folder for storing third record entries in a file." Kistler discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kistler at p. 14. Kistler discloses Vice is a *global server* that is made up of a number of *servers* that appear to the Virtue clients as one single *server*. Kistler at p. 14.

The fourth element of claim 3 is "means for receiving new record file entries at the first memory means." Kistler discloses receiving new first *record entries* in a file at the *first memory store* at a second moment in time later than the first moment in time. Kistler at p. 19. Kistler discloses a client may update cached objects at a later time. Kistler at p. 19.

The fifth element of claim 3 is "means for electronically synchronizing the first folder and the third folder such that a new third record entry is created in the third folder based on the new first record entries." Kistler discloses electronically *synchronizing* the first folder and the third folder. Kistler at p. 18. Kistler discloses that when an object is updated on a client, the updated object is synchronized to the global server, Vice. Kistler at p. 18.

The last element of claim 3 is "means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry." Kistler discloses electronically *synchronizing* the third folder and the second folder. Kistler at p. 18-19. Kistler discloses the *global server*, Vice, informs clients when their objects are outdated and then

the *global server* and the outdated client *synchronizes* so the client gets the updated objects. Kistler at p. 18-19.

4. Claim 4

Claim 4 of the '231 patent reads as follows:

4. A method, comprising: providing a first memory store including a first folder for storing first record entries in a file;

providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

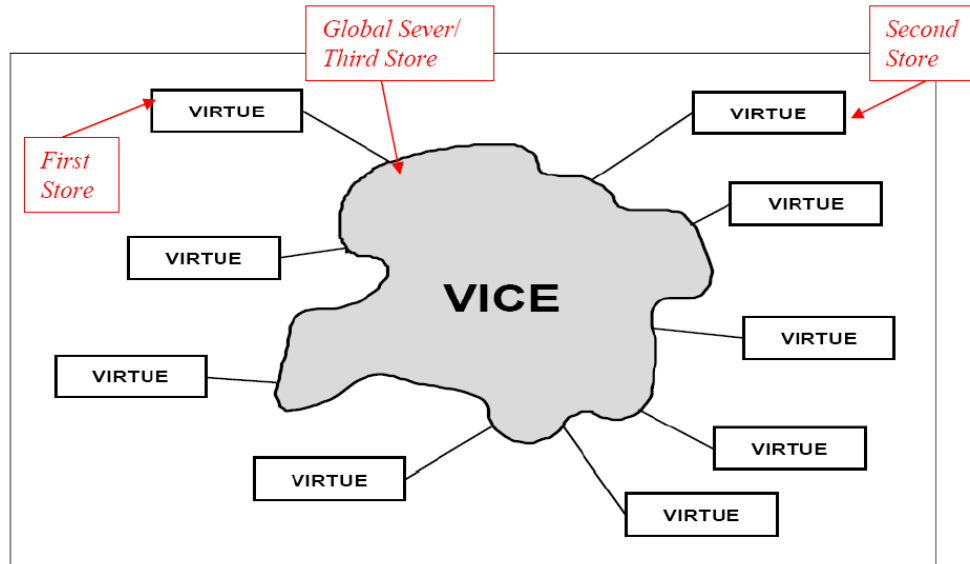
receiving changes at the first memory store;

electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 4 are disclosed in Kistler.

The first element of claim 4 is “providing a first memory store including a first folder for storing first record entries in a file.” Kistler discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kistler at pp. 14, 16-18 and Fig 2.1. Specifically, Kistler discloses that the first data stores are caches on user workstations called Virtue that maintain folders in the file system. Kistler at pp. 14, 16 and Fig. 2.1. Kistler also discloses storing *record entries* in a file since only an object’s descriptors can be stored instead of the entire object. Kistler at pp 17-18.



The structure labeled "Vice" is a collection of trusted servers and untrusted networks. The nodes labeled "Virtue" are private or public workstations (or occasionally time-sharing systems). Vice is reputed to stand for "Vast Integrated Computing Environment," and Virtue for "Virtue is Realized through Unix and Emacs."

Kistler Fig. 2.1 at pg. 14.

The second element of claim 4 is "providing a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time." Kistler also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kistler at pp. 14, 16-19 and Fig 2.1. Kistler discloses that multiple Virtue clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kistler at pp. 14, 16 and Fig 2.1. In addition the second client is also able to store a folder with *record entries* in a file since it can store an object's descriptors instead of the entire object. Kistler at 17-18. Finally, the system is able to determine when one client has updated its cache and in order to notify the other clients to *synchronize* by updating their caches as well. Kistler at p. 19.

The third element of claim 4 is "providing a global server including a third memory store having a third folder for storing third record entries in a file." Kistler discloses providing a *global server* including a *third memory store* having a third folder

for storing third *record entries* in a file. Kistler at p. 14. Kistler discloses Vice is a *global server* that is made up of a number of *servers* that appear to the Virtue clients as one single *server*. Kistler at p. 14.

The fourth element of claim 4 is “receiving changes at the first memory store.” Kistler discloses receiving changes at the *first memory store*. Kistler at p. 19. Kistler discloses a client may update cached objects at a later time. Kistler at p. 19.

The fifth element of claim 4 is “electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Kistler discloses electronically *synchronizing* the first folder and the third folder. Kistler at p. 18. Kistler discloses that when an object is updated on a client, the updated object is synchronized to the global server, Vice. Kistler at p. 18.

The last element of claim 4 is “electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.” Kistler discloses electronically *synchronizing* the third folder and the second folder. Kistler at p. 18-19. Kistler discloses the *global server*, Vice, informs clients when their objects are outdated and then the *global server* and the outdated client *synchronizes* so the client gets the updated objects. Kistler at p. 18-19.

5. Claim 5

Claim 5 of the ‘231 patent reads as follows:

5. A system, comprising: a first memory store including a first folder for storing first record entries in a file;

a second memory store coupled to the first memory store, the second memory store including a second folder for storing second record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory store having a third folder for storing third record entries in a file;

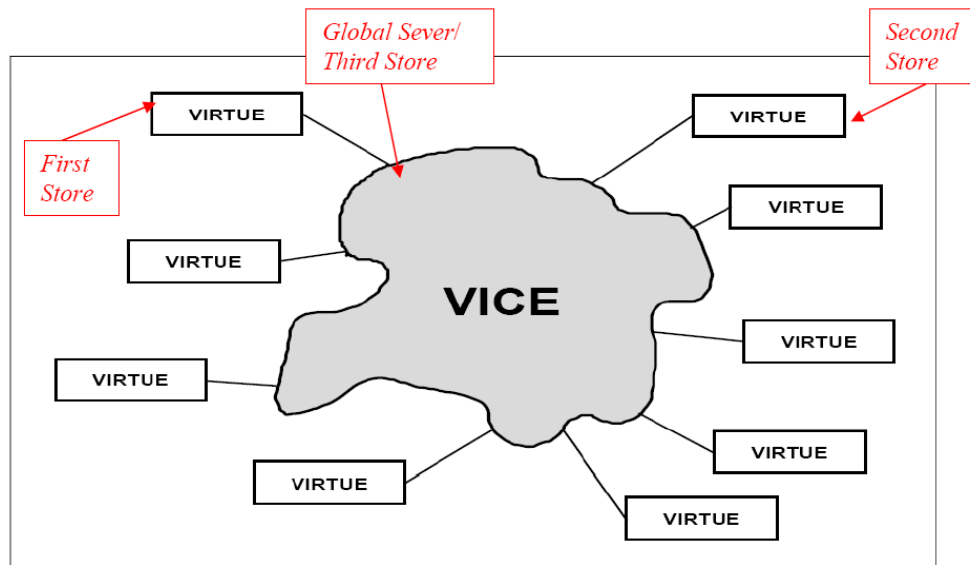
an interface for receiving changes at the first memory store;

a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry is created in the second folder based on the third record entry.

Each of the elements in claim 5 are disclosed in Kistler.

The first element of claim 5 is “a first memory store including a first folder for storing first record entries in a file.” Kistler discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kistler at pp. 14, 16-18 and Fig 2.1. Specifically, Kistler discloses that the first data stores are caches on user workstations called Virtue that maintain folders in the file system. Kistler at pp. 14, 16 and Fig. 2.1. Kistler also discloses storing *record entries* in a file since only an object’s descriptors can be stored instead of the entire object. Kistler at pp 17-18.



The structure labeled "Vice" is a collection of trusted servers and untrusted networks. The nodes labeled "Virtue" are private or public workstations (or occasionally time-sharing systems). Vice is reputed to stand for "Vast Integrated Computing Environment," and Virtue for "Virtue is Realized through Unix and Emacs."

Kistler Fig. 2.1 at pg. 14.

The second element of claim 5 is “a second memory store coupled to the first memory store, the second memory store including a second folder for storing second

record entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time.” Kistler also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kistler at pp. 14, 16-19 and Fig 2.1. Kistler discloses that multiple Virtue clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kistler at pp. 14, 16 and Fig 2.1. In addition the second client is also able to store a folder with *record entries* in a file since it can store an object’s descriptors instead of the entire object. Kistler at 17-18. Finally, the system is able to determine when one client has updated its cache and in order to notify the other clients to *synchronize* by updating their caches as well. Kistler at p. 19.

The third element of claim 5 is “providing a global server including a third memory store having a third folder for storing third record entries in a file.” Kistler discloses providing a *global server* including a *third memory store* having a third folder for storing third *record entries* in a file. Kistler at p. 14. Kistler discloses Vice is a *global server* that is made up of a number of *servers* that appear to the Virtue clients as one single *server*. Kistler at p. 14.

The fourth element of claim 5 is “an interface for receiving changes at the first memory store.” Kistler discloses receiving changes at the *first memory store*. Kistler at p. 19. Kistler discloses a client may update cached objects at a later time. Kistler at p. 19.

The fifth element of claim 5 is “a general synchronization module for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Kistler discloses electronically *synchronizing* the first folder and the third folder. Kistler at p. 18. Kistler discloses that when an object is updated on a client, the updated object is synchronized to the global server, Vice. Kistler at p. 18.

The last element of claim 5 is “a synchronization agent for electronically synchronizing the third folder and the second folder such that a new second record entry

is created in the second folder based on the third record entry.” Kistler discloses a synchronization agent for electronically *synchronizing* the third folder and the second folder. Kistler at p. 18-19. Kistler discloses the *global server*, Vice, informs clients when their objects are outdated and then the *global server* and the outdated client *synchronizes* so the client gets the updated objects. Kistler at p. 18-19.

6. Claim 6

Claim 6 of the ‘231 patent reads as follows:

6. A system, comprising: first memory means including a first folder for storing first record entries in a file;

second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time;

providing a global server including a third memory means having a third folder for storing third record entries in a file;

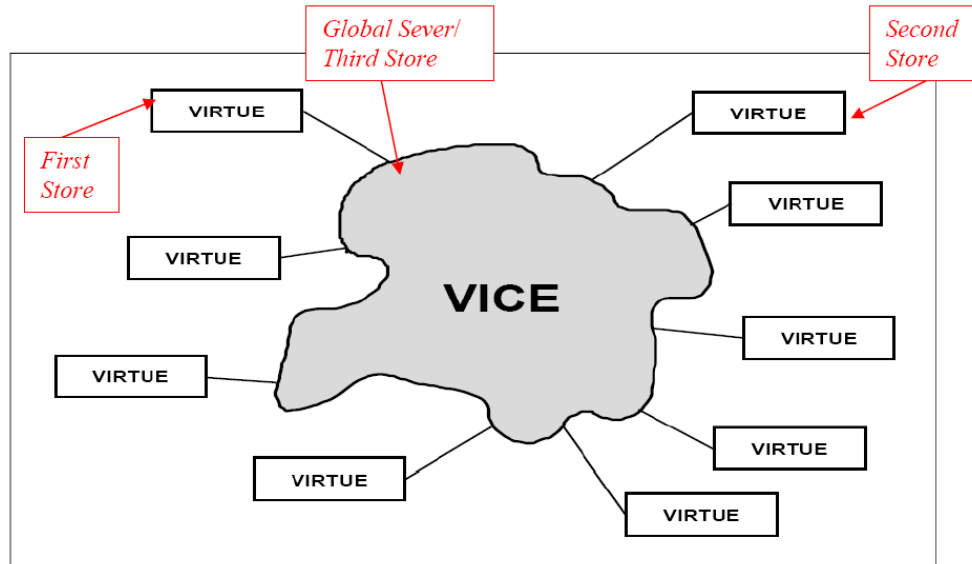
means for receiving changes at the first memory means;

means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry; and

means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.

Each of the elements in claim 6 are disclosed in Kistler.

The first element of claim 6 is “first memory means including a first folder for storing first record entries in a file.” Kistler discloses providing a *first memory store* including a first folder for storing first *record entries* in a file. Kistler at pp. 14, 16-18 and Fig 2.1. Specifically, Kistler discloses that the first data stores are caches on user workstations called Virtue that maintain folders in the file system. Kistler at pp. 14, 16 and Fig. 2.1. Kistler also discloses storing *record entries* in a file since only an object’s descriptors can be stored instead of the entire object. Kistler at pp 17-18.



The structure labeled "Vice" is a collection of trusted servers and untrusted networks. The nodes labeled "Virtue" are private or public workstations (or occasionally time-sharing systems). Vice is reputed to stand for "Vast Integrated Computing Environment," and Virtue for "Virtue is Realized through Unix and Emacs."

Kistler Fig. 2.1 at pg. 14.

The second element of claim 6 is "second memory means coupled to the first memory means the second memory means including a second folder for storing second file entries in a file, the second folder being synchronized with the first folder at a first moment in time such that a content of the second folder is consistent with the content of the first folder at the first moment in time." Kistler also discloses a *second memory store* coupled to the *first memory store*, the *second memory store* including a second folder for storing second *record entries* in a file. Kistler at pp. 14, 16-19 and Fig 2.1. Kistler discloses that multiple Virtue clients can connect to the Coda system so the second store is a second client with a cache manager, Venus, to store data. Kistler at pp. 14, 16 and Fig 2.1. In addition the second client is also able to store a folder with *record entries* in a file since it can store an object's descriptors instead of the entire object. Kistler at 17-18. Finally, the system is able to determine when one client has updated its cache and in order to notify the other clients to *synchronize* by updating their caches as well. Kistler at p. 19.

The third element of claim 6 is "providing a global server including a third memory means having a third folder for storing third record entries in a file." Kistler discloses providing a *global server* including a *third memory store* having a third folder

for storing third *record entries* in a file. Kistler at p. 14. Kistler discloses Vice is a *global server* that is made up of a number of *servers* that appear to the Virtue clients as one single *server*. Kistler at p. 14.

The fourth element of claim 6 is “means for receiving changes at the first memory means.” Kistler discloses receiving changes at the *first memory store*. Kistler at p. 19. Kistler discloses a client may update cached objects at a later time. Kistler at p. 19.

The fifth element of claim 6 is “means for electronically synchronizing the first folder and the third folder such that the content of the changes creates a third record entry.” Kistler discloses electronically *synchronizing* the first folder and the third folder. Kistler at p. 18. Kistler discloses that when an object is updated on a client, the updated object is synchronized to the global server, Vice. Kistler at p. 18.

The last element of claim 6 is “means for electronically synchronizing the third folder and the second folder, such that a new second record entry is created in the second folder based on the new third record entry.” Kistler discloses electronically *synchronizing* the third folder and the second folder. Kistler at p. 18-19. Kistler discloses the *global server*, Vice, informs clients when their objects are outdated and then the *global server* and the outdated client *synchronizes* so the client gets the updated objects. Kistler at p. 18-19.

N. Claims 1-6 are Rendered Obvious by the Coda System Under 35 U.S.C. § 103

Please see attached Exhibit CC-J, CC-K and CC-L presenting claim charts for comparison of the Lotus Notes System with claims 1-6 of the ‘231 patent.

Although Requester believes each Coda reference individually anticipates all claims of the ‘231 patent, in event the Examiner does not find claims 1-6 anticipated by the Coda references individually, Requester alternatively submits that claims 1-6 of the ‘231 patent are rendered obvious by Satyanarayanan in view of Kumar further in view of Kistler under 35 U.S.C. § 103. Claim charts applying Satyanarayanan in view of Kumar further in view of Kistler to these claims are submitted herewith as Exhibits CC-N.

All three references describe with particularity the Coda system developed at Carnegie Mellon University for *synchronizing* data between *server* and client computers. It would have been obvious to one of ordinary skill in the art to combine the teaching in all three references relating to the Coda system together in order to get a fuller understanding of the Coda system.

Claims 1-6 are rendered obvious for the same reasons as given above for the Coda references individually anticipating the claims.

IV. EXPECTED ARGUMENTS AND AMENDMENTS

Currently, four patents assigned to the Patent Owner with similar disclosure of the '231 are undergoing reexamination, including the '221 patent to which the '231 claims priority. The four patents with control numbers for the reexaminations are as follows:

US Patent No 6,708,221	Control Nos. 90/008,131 and 90/008,162
US Patent No 6,085,192	Control No. 90/008,062
US Patent No 6,023,708	Control Nos. 90/007,421 and 90/007,933
US Patent No 7,039,679	Control No. 90/008,397

Patent Owner has made various arguments and amendments in the reexaminations that it is likely to raise again in a reexamination of the '231 patent. Since requestor is not allowed to later comment in an ex parte proceeding, it is necessary for requestor to address Patent Owner's arguments in this request.

Examples of typical responses by the Patent Owner include amending "translation" and "smart phone" into to the claims and arguments on how Lotus Notes or other asserted prior art is not capable of *translation* or running on a *smart phone*, amending or adding claims to recite only *synchronizing* on a detected change or predetermined number of changes, and arguments that neither Lotus Notes nor the other art of record disclose *synchronizing* changes. Other examples include Patent Owner amending or adding claims to recite a *firewall* limitation along with arguments that Lotus Notes and the other asserted prior art does not disclose a *firewall*, as well as arguments to narrow the definition of certain claim terms such as *firewall* and *translation*.

A. Lotus Notes Running on a *Smart Phone*

Patent Owner has repeatedly asserted that Lotus Notes is incapable of being run on a *smart phone* in the relevant time frame. An example of their argument is presented below.

“The Declaration of Steven Beckhardt at paragraph 7 explains that the Lotus Notes software (described by Brown and Lamb) required a PC based operating system. Smart phones in the relevant time frame could not support such operating systems. For example, Lotus Notes employs algorithms necessary for replication in which an index of documents is built, requiring memory resources consistent with a PC, such a configuration on a smart phone was not possible.”

Patent Owner Response pg. 14 of Reexamination Control No. 90/008,397.

This statement is in direct contradiction to the argument presented by the Patent Owner during prosecution of the ‘221 patent in order to overcome a §112 rejection that a *smart phone* was not enabled by the specification due to its inherent inability to run desktop computer applications.

“For Example, the examiner bases his rejection of claims 86 and 93 on the proposition that limited memory or processing power of smart phones or settop boxes renders them incapable of running desktop computer applications. However, in even a general sense, it was common knowledge at the time of invention that portable version of PC-based operating systems were being developed, such as Windows CE, for running applications on portable devices. Moreover, it was necessarily within the knowledge and skill of ALMOST ANYONE able to secure work networking at the time of the invention to otherwise implement general storing, determining, and sending using applicable devices, and the specification need not include the manual for any, let alone every such device.”

‘221 File Wrapper, paper #11, Patent Owner’s Amendment and Response dated 12-30-2002. (emphasis in original)

To get their claims allowed during prosecution, Patent Owner asserted it was within the skill of the art of almost anyone to get desktop computer applications to run on a *smart phone*. Yet, now in reexamination Patent Owner is asserting that it was not possible to run a desktop computer application like Lotus Notes on a *smart phone*. The

admission of the Patent Owner during prosecution to get their claims allowed is binding on the Patent Owner during reexamination. MPEP § 2217(III).

B. Prior Art Disclosure of *Smart Phone*

Patent Owner has also asserted Lamb does not disclose a *smart phone*. See Patent Owner Response pg. 13-14 of Reexamination Control No. 90/008,397. Patent Owner points out that the disclosure that Requester asserts teaches a *smart phone* is contained in a chapter entitled “What the Future Holds.” However, Patent Owner fails to mention this chapter also discusses what the state of technology was at the time Lamb was published. An example is the below except of Lamb’s disclosure of *smart phone*:

“Many of the PCMCIA credit-card-sized modems allow attachment to a cellular phone. With today’s technology, you can have a 3-pound notebook computer with a Pentium or Power PC processor, a 28.8Kbps PCMCIA modem with cellular phone capabilities, and Lotus Notes installed. This will give good groupware access performance for the most mobile of users. And that’s with today’s technology.” Lamb at 196. (emphasis added)

Lamb therefore discloses the use Lotus Notes running on a *smart phone* in the form of a laptop capable of being used as a computer or as phone through the directly attached cell phone. Alternatively, the disclosure of using Lotus Notes on a PDA, such as a Apple Newton, makes the use of Lotus Notes on a smart phone obvious under 35 USC § 103. Lamb at 196.

In addition to Lamb, other prior art of record disclose smart phone. For example, GloMop discloses a data *synchronization* system run on a HP100LX smart phone. GloMop at p. 6, § 1.8. Another example is disclosed in Lambert which utilizes clients that are of limited resources and infrequently connected to the network. Lambert at 1. Although Requester believes these references anticipate a *smart phone* limitation, in the alternative they would a *smart phone* limitation obvious.

C. Lotus Notes *Translation*

Patent Owner has repeatedly asserted that Lotus Notes is incapable of *translation*. For example, Patent Holder argued “translation of different document formats was not supported by Lotus Notes in the relevant time frame.” Patent Owner Response pg. 56 of

Reexamination Control Nos. 90/007,421 and 90/007,933. This argument was supported by a declaration from Stephen Beckhardt, stating:

A replication between a 3.0 Notes server and a 4.0 Notes server would not be considered a translation between different document formats. This is because such a replication would address relatively minor differences in network or disk storage formats, but not substantive translations between different document formats.

Beckhardt Declaration, Patent Owner Response of Reexamination Control Nos. 90/007,421 and 90/007,933.

However, Lotus Notes discloses *translation*. In addition to disclosures of Lotus Notes *translation* identified in reexaminations on Visto's '708 and '679 patents, Stephen Beckhardt testified that Judge Ward's construction of *translation* in *Visto Corp. v. Seven Networks, Inc.* included differences in network or disk storage formats—*translation* that occurs between 3.0 Notes and 4.0 Notes:

Q: ...The Court adopts Visto's definition of translator to mean software routines or code that convert information or data in one format to information or data in a second format. Is this your understanding that that is broader than just a change in the data model?

A: Yes.

Q: Okay. Would it encompass, for example, changing data from one of the three formats you described in your report where it says the on-disk structure or the wire structure or in memory format to another one of those formats?

A: Yes. The Court's definition of changing a format would encompass changing from on-disk to in memory to on the network.

Exhibit OTH-J, Stephen Beckhardt's Deposition, pgs. 47-48.

Moreover, Stephen Beckhardt testified that there were substantive differences between 3.0 Notes and 4.0 Notes formats:

Q. There was - - Was there any difference between the way data was stored in release 3 and release 4?

A. Yes.

Q. Okay. And what was that difference?

A. An example that is referred to I believe in the Balaban report. Version 4 stored information that encoded a time stamp on each field of a document, so that the version 4 Notes database system kept track of when each field of a document changed as opposed to version 3, which only kept track of when the entire document was changed as a whole.

Exhibit OTH-J, Stephen Beckhardt's Deposition, pgs. 46-47.

These substantive format differences require translation when Lotus Notes replicates between 3.0 Notes and 4.0 Notes databases.

D. LOTUS NOTES *SYNCHRONIZATION*

Patent Holder attempted to overcome Lotus Notes prior-art by arguing that Lotus Notes sends a database replica during replication: "the hub server replicates with each workstation, in a serial fashion Once this process is concluded, the final database replica is provided to Hub X from Server A....The operation of Lotus Notes is point to point. Replication is always between two points, thus, the Lotus notes art does not anticipate the issued claims." (emphasis added). Patent Owner Response pg. 25 of Reexamination Control No. 90/008,131 and 90/008,162. However, Patent Holder cannot overcome Lotus Notes prior-art by arguing Lotus Notes does not *synchronize* data in the proper manner. Visto argued that the term "synchronizing" should be construed to mean, "providing for data consistency by reconciling modifications to stored information." See Exhibit OTH-C, Joint Claim Construction and Pre-Hearing Statement in *Visto Corp. v. Research in Motion, Ltd.*, Civil Action No. 2:06-CV-181-TJW (E.D. Tex. filed September 4, 2007). The Lotus Notes serial point to point replication *synchronizes* the first device with the global server and independently *synchronizing* the second device with the *global server* so that the changes are indirectly transferred between the first and second device. Therefore, under Visto's proposed construction for the term "synchronizing," Lotus Notes *synchronizes* data. Furthermore, the serial point-to-point replication of Lotus Notes is the same as the discussion of synchronizing data indirectly in the '231 patent excerpt below.

"Accordingly, the synchronization means independently synchronizes the selected portions of workspace data 116 [on the remote device] and 136 [on the LAN device] with the selected portions of the workspace data 120

[on the global server]. Thus, the synchronization means indirectly synchronizes workspace data 136 [on the LAN device] with workspace data 116 [on the remote device].”

‘231 patent at Col. 5, lines 5-9.

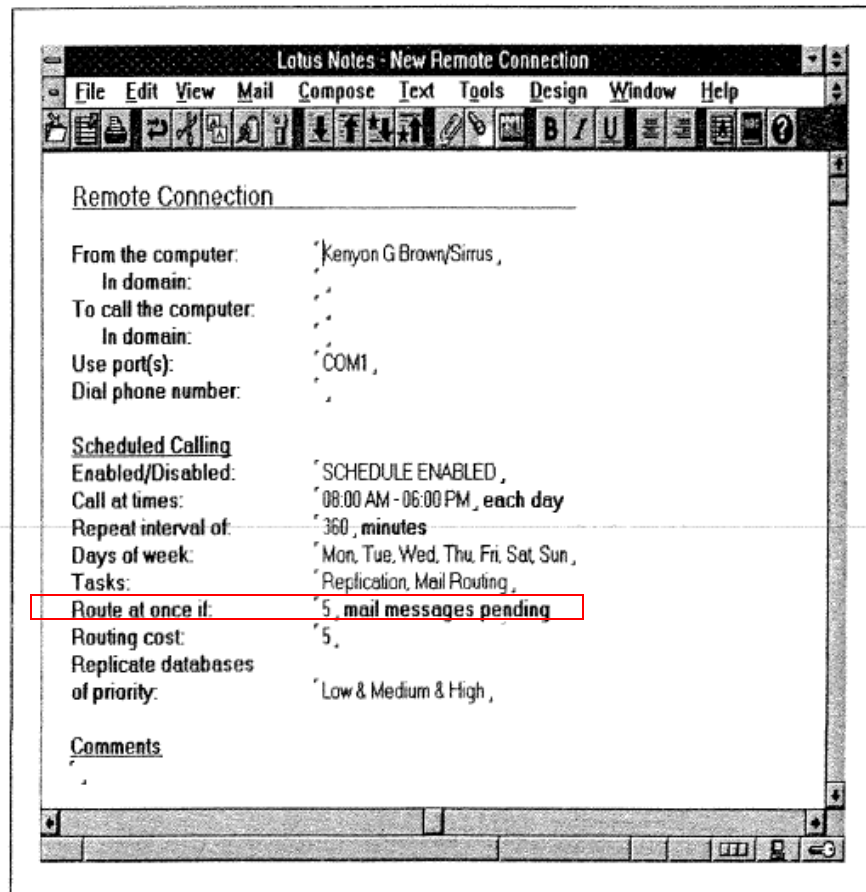
Patent Holder further supported its argument, stating “Patent Holder notes that the provision of the winning replica to the second Lotus workstation is not sending the differences to the second workstation. This is because, when a winner is declared, that replica is sent; the differences between replicas will be lost upon selection of one replica over another.” Patent Owner Response pg. 26 of Reexamination Control No. 90/008,131 and 90/008,162. But, for clarification, during replication Lotus Notes does not simply send a winning replica. As described in Brown, Lamb, and other Lotus Notes references, Lotus Notes replicates modified folders and/or documents within a replica (Brown at pp. 8, 40, 571) or the modified fields of documents within a replica. Lamb at 28. Thus, Lotus Notes *synchronizes* the changes to “record entries” and Visto cannot overcome Lotus Notes prior-art.

E. *Synchronizing on a Detected Change or Predetermined Number of Changes*

The Patent Owner also asserts that none of the prior art shows *synchronizing* on a detected change or a predetermined number of changes. However, the prior art of record does disclose these elements. For instance Brown and Lamb clearly disclose that Lotus Notes has the capability of *synchronizing* after a predetermined number of new emails have been created by allowing the user to “[s]pecify the number of outgoing mail messages that automatically trigger a call to the server.” Brown at 208, *See id.* at 206, Figure 10.1 (“route at once if 5 mail messages pending”). See also Lamb at Fig 3.1, 3.2, and 3.3.

FIGURE 10.1

The Remote Connection document

**FIGURE 10.1 OF BROWN**

In addition at least Lambert and GloMop also disclose *synchronizing* on a detected change.

“The Pcmail design therefore allows two modes of communication between repository and client. ‘Interactive mode’ is used when the client is always connected to the network. Any changes to the client's local mail state are immediately also made to the repository's global mail state, and any incoming mail is immediately transmitted from repository to client.

Lambert at 3. (emphasis added)

“Mobile applications can check the file cache before asking for information from the proxy. The client can ask the proxy to verify that the local version of a given file is current.”

GloMop at 2.1. (The client cache synchronizes with the proxy server when it detects an updated version of a file.)

F. Firewall

Another common amendment by the Patent Owner is to include a limitation regarding *firewalls*. Patent Owner has attempted to overcome the prior art by reading a very narrow definition of firewall into the claims. For example, Patent Owner argued that “[n]aturally, a user ID/Password exchange cannot be considered a LAN firewall by any stretch of the imagination.” See Patent Owner Response pg. 10 of Reexamination Control No. 90/008,397. In addition, Patent Holder offered the declaration of Steven Beckhardt in support of its argument that “Notes based authentication would not have solved the problem of the patent subject to this reexamination proceeding”:

That one of skill in the art would not recognize Lotus Notes server authentication as being a firewall as recited in the claims. A firewall is utilized to prevent improper network access from occurring at all; server authentication is only possible after network access is achieved. The purpose of Lotus Notes authentication is to verify that a user accessing a server has a confirmed identity. Once this is performed, access (control) verification will determine which, if any, databases of the server are accessible by the identified user. Conversely, a firewall typically operates at the packet and port level of a network, preventing improper connections from being made at all.

Patent Owner Response pg. 24 of Reexamination Control No. 90/008,062 and attached Beckhardt Declaration.

However, Patent Holder has abandoned its narrow construction in litigation. After asserting the narrow firewall construction in early litigation, Visto asked for the *Visto Corp. v. Seven Networks, Inc.* Court’s broad construction in the RIM litigation. See Exhibit OTH-C, Copy of Patent Rule 4-3 Joint Claim Construction and Pre-Hearing Statement in *Visto Corp. v. Research in Motion, Ltd.*, Civil Action No. 2:06-CV-181-TJW (E.D. Tex. filed September 4, 2007). Under this construction, Visto’s expert Stephen Beckhardt testified that he agrees Lotus Notes discloses *firewall*:

Q. Okay. There’s no question that Lotus Notes had extensive authentication procedures, correct?

A. That is correct.

Q. So if the court's claim construction was correct [Judge Ward's construction of firewall from Seven litigation], would you agree that Lotus Notes included this element of a firewall?

A. Yes.

Exhibit OTH-J, Stephen Beckhardt's Deposition, pgs, 27-28.

G. TRANSLATION

Patent Holder has claims that require *translation* and also makes amendments adding *translation* to claims of patents in reexamination. The *Visto Corp. v. Seven Networks, Inc.* Court's construction of *translation* requires "Software routines or code that converts information or data in one format to information or data in another format."

To avoid prior-art during reexamination, Patent Holder makes arguments to narrow the definition of translation arguments that contradict positions Patent Holder takes in litigation. Patent Holder argued "that the currently claimed embodiment require [sic] a translator and/or translation function. . . . 'Translation **is needed for synchronization** when a first workspace element is **in a different format** than a second workspace element.'" Patent Owner Response pg. 53 of Reexamination Control No. 90/007,421 and 90/007,933. Patent Holder continued by arguing that the claim scope allowing same format translation "was unequivocally surrendered during prosecution of the '708 patent" on January 19, 1999. Patent Owner Response pg. 53-54 of Reexamination Control No. 90/007,421 and 90/007,933. But in 2006 during litigation, Patent Holder argued for a broader claim scope: the defendant "argues at considerable length that the 'first format' and the 'second format' must necessarily be different formats (citing to various portion of the '708 specification referring to data consistency and translation). . . . These arguments, however, are rebutted by the simple fact that not all the claims require the first format and the second format to be the same." See OTH-K, Visto's Reply Brief in Support of its Claim Construction, pp. 10, *Visto v. Sproquit Technologies*, Civil Action No. 3:04-cv-00651-EMC (N.D. Cal. June 16, 2006). Accordingly, under the construction urged by Patent Owner in litigation, the Office must

continue to construe the *translation* term to allow formats to change in accordance with Patent Owner's claim construction.

If Patent Owner seeks to narrow its claims, it is required to amend its claims in reexamination to specifically recite the narrowing language. In addition, Patent Owner must still show it can satisfy 35 USC § 112 with regards to any newly added claim elements.

V. CONCLUSION

The prior art documents presented in the above Request were either not previously considered by the Office or are now being presented in a new light pursuant to MPEP § 2242(II)(A). The claims of the '231 patent are not patentable over the prior art documents cited herein. The prior art documents teach the subject matter of the '231 patent in a manner such that substantial new questions of patentability for all claims are raised by this Request.

In view of the foregoing, it is respectfully submitted that a substantial new question of patentability of claims 1-6 of U.S. Patent No. 7,225,231 has been raised by this Request. Accordingly, the Office is requested to grant this Request and to initiate reexamination with special dispatch.

As an aid to the application of the presented prior art to claims of the '231 patent, corresponding claim charts are provided at Exhibit CC-A through CC-N attached hereto.

Enclosed is a credit card authorization to cover the Fee for reexamination. If this authorization is missing or defective please charge the Fee to the Novak Druce Deposit Account No. 14-1437.

Respectfully submitted,

A handwritten signature in dark ink, appearing to read "Tracy W. Druce", is written over a horizontal line.

Novak Druce & Quigg, LLP

Donald J. Quigg

Reg. No. 16,030

Tracy W. Druce

Reg. No. 35,493

James P. Murphy

Reg. No. 55,474

Attorneys for Third Party Requester

Research in Motion. Ltd.

NOVAK DRUCE & QUIGG, LLP

1000 Louisiana Ave.

53rd Floor

Houston, Texas 77002

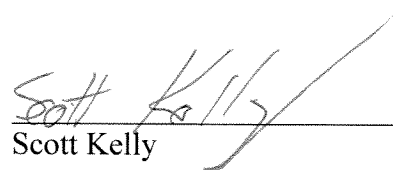
P: 713-571-600

F: 713-456-2836

CERTIFICATE OF SERVICE

The undersigned hereby certifies that a copy of this Request for *Ex Parte* Reexamination, together with all exhibits and attachments and supporting documentation, has been served via priority mail on the 9th day of January, 2008, upon the following:

SCOTT MCKEOWN
OBLON, SPIVAK, MCCLELLAND MAIER &
NEUSTADT, P.C.
1940 DUKE STREET
ALEXANDRIA VA 22314


Scott Kelly